Foreground-aware Semantic Representations for Image Harmonization

Konstantin Sofiiuk
k.sofiiuk@samsung.com

Polina Popenova
p.popenova@partner.samsung.com

Anton Konushin
a.konushin@samsung.com

Samsung AI Center – Moscow

Abstract

Image harmonization is an important step in photo editing to achieve visual consistency in composite images by adjusting the appearances of a foreground to make it compatible with a background. Previous approaches to harmonize composites are based on training of encoder-decoder networks from scratch, which makes it challenging for a neural network to learn a high-level representation of objects. We propose a novel architecture to utilize the space of high-level features learned by a pre-trained classification network. We create our models as a combination of existing encoder-decoder architectures and a pre-trained foreground-aware deep high-resolution network. We extensively evaluate the proposed method on the existing image harmonization benchmark and set up a new state-of-the-art in terms of MSE and PSNR metrics. The code and trained models are available publicly.

1. Introduction

The main challenge of image compositing is to make the output image look realistic, given that the foreground and background appearances may differ greatly due to photo equipment specifications, brightness, contrast, etc. To address this challenge, image harmonization can be used to make those images visually consistent. In general, image harmonization aims to adapt the appearances of the foreground region of an image to make it compatible with the new background, as can be seen in Fig. 1.

In recent years, several deep learning-based algorithms have been addressed to this problem [34, 6, 5, 12]. Unlike traditional algorithms that use handcrafted low-level features [14, 18, 33, 40], deep learning algorithms can focus on the image contents.

For image harmonization, it is crucial to understand what the image foreground and background is and how they should be semantically connected. For example, if the to-be-harmonized foreground object is a giraffe, it is natural to adjust the appearance and color to be blended with surrounding contents, instead of making the giraffe white or red. Therefore, Convolutional Neural Networks (CNNs) have succeeded in such tasks, showing an excellent ability to learn meaningful feature spaces, encoding diverse information ranging from low-level features to high-level semantic content [17, 43].

In recent papers on image harmonization, models are trained from scratch using only annotations that do not contain any semantic information [6, 5]. The neural network is supposed to learn all dependencies between semantic patterns without supervision. However, it proves to be an extremely difficult task even for deep learning. The model is more likely to fall into a local minimum with relatively low-level patterns rather than learn high-level abstractions due to several reasons. First, learning semantic information using this approach is similar to self-supervised learning, where the task of automatic colorization is solved [20, 19, 44]. However, several works focusing on self-supervised learning demonstrate that the resulting representations are still inferior to those obtained by supervised learning on ImageNet [17, 9]. Second, the amount of data used for image harmonization training is by orders of magnitude smaller than the ImageNet dataset [7] and other datasets used for self-supervised training. Considering all the aforementioned aspects, it seems challenging for a neural network to learn high-level semantic features from scratch only during image harmonization training. Tsai *et al*. [34] also highlighted this challenge and proposed a special scene parsing decoder to predict semantic segmentation, although it provided only an insignificant increase in quality and required semantic segmentation annotation of all training images. Consequently, this technique was not used in the recent papers on this topic [6, 5].

In this paper, we propose a simple approach to the effective usage of high-level semantic features from models pre-trained on the ImageNet dataset for image harmonization. We find that the key factor is to transfer the image and the corresponding foreground mask to a pre-trained model. To achieve such transfer, we present a method of adapting the network to take the image and the corresponding foreground mask as the input without negatively affect-

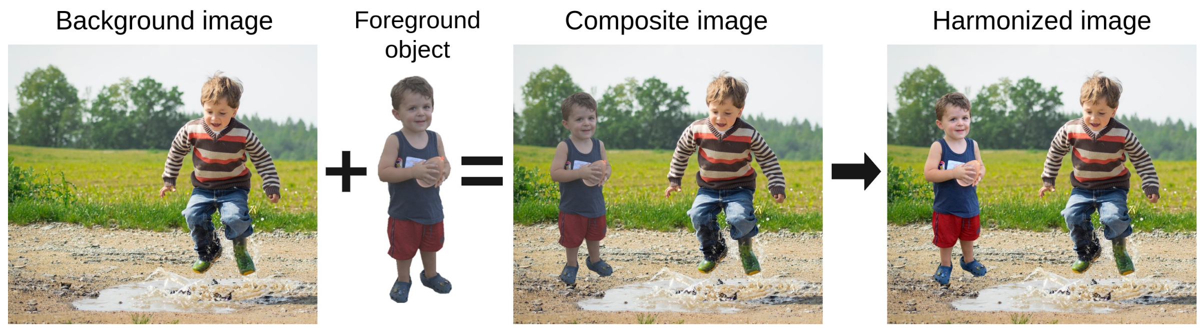

Figure 1: Illustration of the practical application of image harmonization algorithm to a composite image. Best view in color.

ing the pre-trained weights. In contrast to previous works, we make existing pre-trained models foreground-aware and combine them with encoder-decoder architectures without adding any auxiliary training tasks.

Another challenging task is to create valid training and test datasets, since producing a large dataset of real harmonized composite images requires a lot of human labor. For this reason, Tsai *et al*. [34] proposed a procedure for creating synthesized datasets, although their dataset was not published. Cong *et al*. [5] reproduced this procedure, added new photos to the dataset and made it publicly available. To evaluate the proposed method, we conduct extensive experiments on this synthesized dataset and perform a quantitative comparison with previous algorithms.

2. Related work

In this section, we review image harmonization methods and some related problems as well.

Image harmonization. Early works on image harmonization use low-level image representations to adjust foreground to background appearance. Existing methods apply alpha matting [32, 35], gradient-domain methods [27, 14], matching color distribution [29, 4, 28] and multi-scale statistics [33]. Combinations of these methods are used to assess and improve realism of the images in [18, 40].

Further work on image realism was provided by Zhu *et al*. [48]. They fit a CNN model to distinguish natural photographs from automatically generated composite images and adjust the color of the masked region by optimizing the predicted realism score. The first end-to-end CNN for image harmonization task was proposed by Tsai *et al*. [34]. Their Deep Image Harmonization (DIH) model exploits a well-known encoder-decoder structure with skip connections and an additional branch for semantic segmentation. The same basic structure is broadly used in related computer vision tasks such as super-resolution [45], image colorization [44, 10], and image denoising [24]. Cun *et al*. [6] also go with an encoder-decoder U-Net-based [30] model adding spatial-separated attention blocks to the decoder.

Standard encoder-decoder architectures with a content loss can be supplemented by an adversarial loss too. Usually, these models are successfully trained with no use of the GAN structure, but in some cases adversarial learning makes an impact on image realism. The approach can be found in the papers on super-resolution [21, 37] and image colorization [15, 26]. Cong *et al*. [5] construct an encoder-decoder model based on the Deep Image Harmonization architecture [34] with spatial-separated attention blocks [6]. Besides the classic content loss between the harmonized and target image, they also add the adversarial loss from two discriminators. While the global discriminator predicts whether the given image is fake or real as usual, the domain verification discriminator checks if the foreground and background areas are consistent with each other.

Image-to-image translation. Image harmonization can be regarded as an image-to-image translation problem. Some GAN architectures are designed for such tasks. Isola *et al*. [12] describe a pix2pix GAN, which is trained to solve image colorization and reconstruction problems among other tasks, and can be applied to image harmonization. There are several GAN models for the related task of image blending [38, 42], which seamlessly blend object and target images coming from different sources.

Existing general image-to-image translation frameworks are not initially designed for the image harmonization task and do not perform as well as specialized approaches. Cun *et al*. [6] present the results for both dedicated model, based on U-Net, and pix2pix model [12], and the latter fails to get competitive metric values.

Single image GANs. In order to generate realistic images, GANs require a lot of training samples. There are recent works on adversarial training with just a single image [31, 11]. The SinGAN and ConSinGAN models do not require a large training set and learn to synthesize images similar to a single sample. It could also be used for unsupervised image harmonization. However, these models show

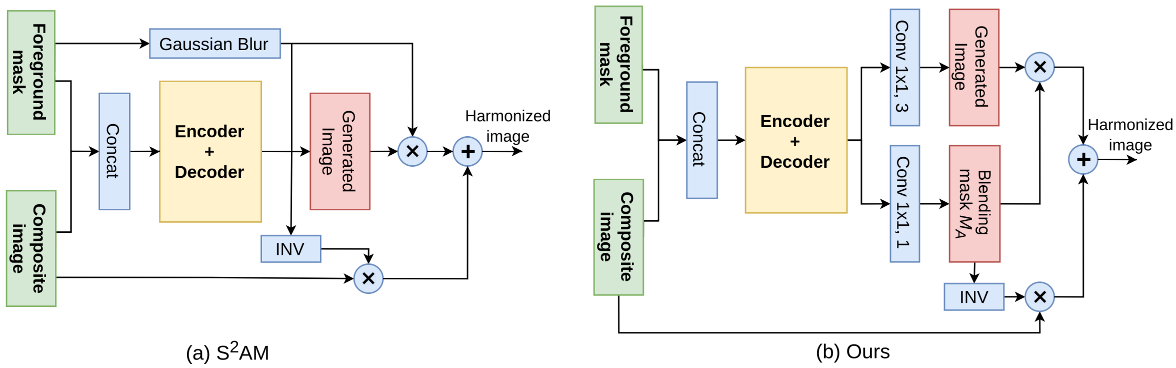

Figure 2: (a) The architecture of the S^{2}AM [6] approach to inference a resulting image. (b) The proposed architecture. We find that a network can easily predict a mask for blending since a foreground mask is fed as an input to the encoder-decoder. No need for heuristics, *e.g*. blurring the foreground mask.

good performance only in cases when the background image contains many texture and stylistic details to be learned (*e.g*. a painting) and the foreground is photorealistic, leading to artistic image harmonization. When it comes to the composite images based on real-world photos, the model may achieve rather poor results. Also, SinGAN models require training from scratch for every image, so harmonization of one composite may take a lot of computational time.

3. Proposed method

In this section, we first revisit encoder-decoder architectures for image harmonization in Sec. 3.1. Then, we introduce a mask fusion module to make pre-trained image classification models foreground-aware in Sec. 3.2 and demonstrate the detailed resulting network architecture in Sec. 3.3. Finally, we present the Foreground-Normalized MSE objective function in Sec. 3.4.

Below we use the following unified notation. We denote an input image by $I \in \mathbb{R}^{H\times W\times 3}$, a provided binary mask of a composite foreground region by $M \in \mathbb{R}^{H\times W\times 1}$, and concatenation of I and M by $\hat{I} \in \mathbb{R}^{H\times W\times 4}$, where H and W are height and width of the image.

3.1. Encoder-decoder networks

We consider image harmonization as an image-to-image problem, with a key feature of mapping a high resolution input to a high resolution output. Many previous works have used an encoder-decoder network [34, 6, 5], where the input is passed through a series of layers that progressively downsample until a bottleneck layer, where the process is reversed. Skip connections between an encoder and a decoder are essential for image harmonization, as they help to avoid image blurring and texture details missing.

In this section, we present our modification of the encoder-decoder network with skip connections that was introduced in DIH [34]. We change the original DIH architecture by removing the fully connected bottleneck layer to make the model fully convolutional.

DIH reconstructs both background and foreground regions of an image, which is not optimal for solving the task of image harmonization, as a background should remain unchanged. Therefore, we propose a simple scheme inspired by S^{2}AM [6], where the network predicts a foreground region and blending mask M_A. The final image is obtained by blending an original image and a decoder output using a predicted blending mask, as shown in Fig. 2. It allows the model to control blending of a predicted foreground region and a background. We provide a visual comparison of input foreground masks and inverted blending masks obtained from the trained iDIH model in Fig. 3.

Figure 3: Blending masks predicted by the iDIH model.

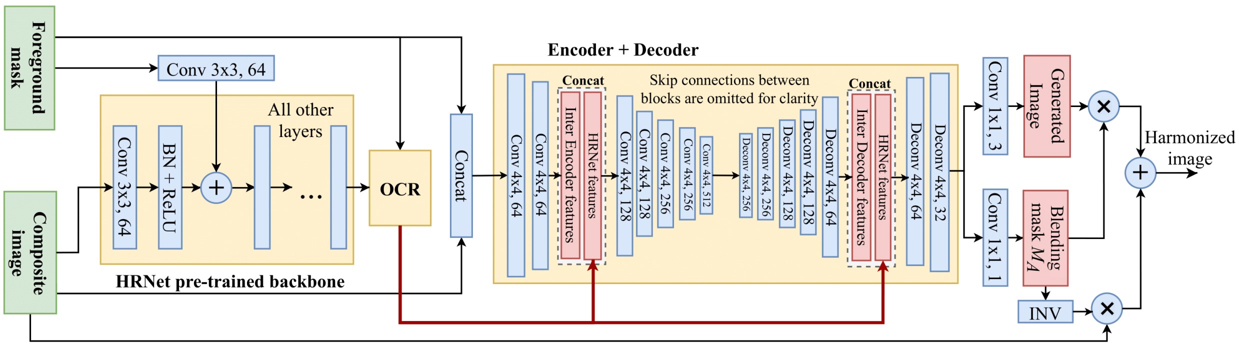

Figure 4: Our final architecture based on HRNet+OCR [41] (iDIH-HRNet).

Let us denote the encoder-decoder network as $D_F(\hat{I}) : \mathbb{R}^{H \times W \times 4} \rightarrow \mathbb{R}^{H \times W \times C}$, which returns features $x = D_F(\hat{I})$ with C channels, and denote two 1×1 convolution layers as $D_{RGB} : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}^{H \times W \times 3}$, $D_M : \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}^{H \times W \times 1}$. The output image can be formalized as:

$$I^{pred} = I \times [1 - D_M(x)] + D_{RGB}(x) \times D_M(x). \quad (1)$$

We refer our modification of the DIH architecture with the above-mentioned enhancements as improved DIH (iDIH).

3.2. Foreground-aware pre-trained networks

In many computer vision domains such as semantic segmentation, object detection, pose estimation, etc., there are successful practices of using neural networks pre-trained on the ImageNet dataset as backbones. However, for image harmonization and other image-to-image translation problems, there is no general practice of using such networks.

Similarly to image-to-image translation, semantic segmentation models should have the same input and output resolution. Thus, semantic segmentation models seem promising for image harmonization, because they typically can produce detailed high-resolution output with a large receptive field [25]. We choose current state-of-the-art semantic segmentation architectures such as HRNet+OCR [36, 41] and DeepLabV3+ [3] as the base for our experiments.

Foreground masks for pre-trained models. Models trained on ImageNet take an RGB image as an input. Without awareness of the foreground mask, the network will not be able to accurately compute specific features for the foreground and the background and compare them with each other, which can negatively affect the quality of prediction. Similar issues arise in interactive segmentation and RGB-D segmentation tasks [39, 1, 8]. The most common solution is to augment the weights of the first convolution layer of a pre-trained model to accept N-channels input instead of only an RGB image. We discover that this solution can be modified by adding an extra convolutional layer that takes the foreground mask as an input and produces 64 output channels (to match conv1 in pre-trained models). Then, its outputs are summarized with the conv1 layer outputs of the pre-trained model. A core feature of this approach is that it allows setting a different learning rate for the weights that process the foreground mask.

3.3. Final architecture design

We extract only high-level features from the pre-trained models: outputs after the ASPP block in DeepLabV3+ [3] and outputs after the OCR module in HRNet [36, 41]. There are several ways to pass the extracted features to the main encoder-decoder block:

- pass them to one fixed position of the encoder (or decoder or both simultaneously);
- build a feature pyramid [22, 36] on them and pass the resulting feature maps to the encoder layers with appropriate resolutions (or the decoder or both simultaneously).

Our experiments have shown that the best results are obtained by passing the features to both encoder and decoder simultaneously, as can be found in Table 6. When passing the features only to the decoder the worst results are obtained. It can be explained by the fact that high-level reasoning in the encoder-decoder happens at the bottleneck point, then the decoder relies on this information and progressively reconstructs the output image. Passing the features to the encoder only leads to the decoder lacking the semantic information from the pre-trained model.

We implement a variant of feature pyramid as in HRNetV2p [36] and find that there is no need to use it to pass the features to the encoder. Without any loss of accuracy, it is sufficient to pass the features to a single position in the encoder and decoder, and the encoder is trained from scratch

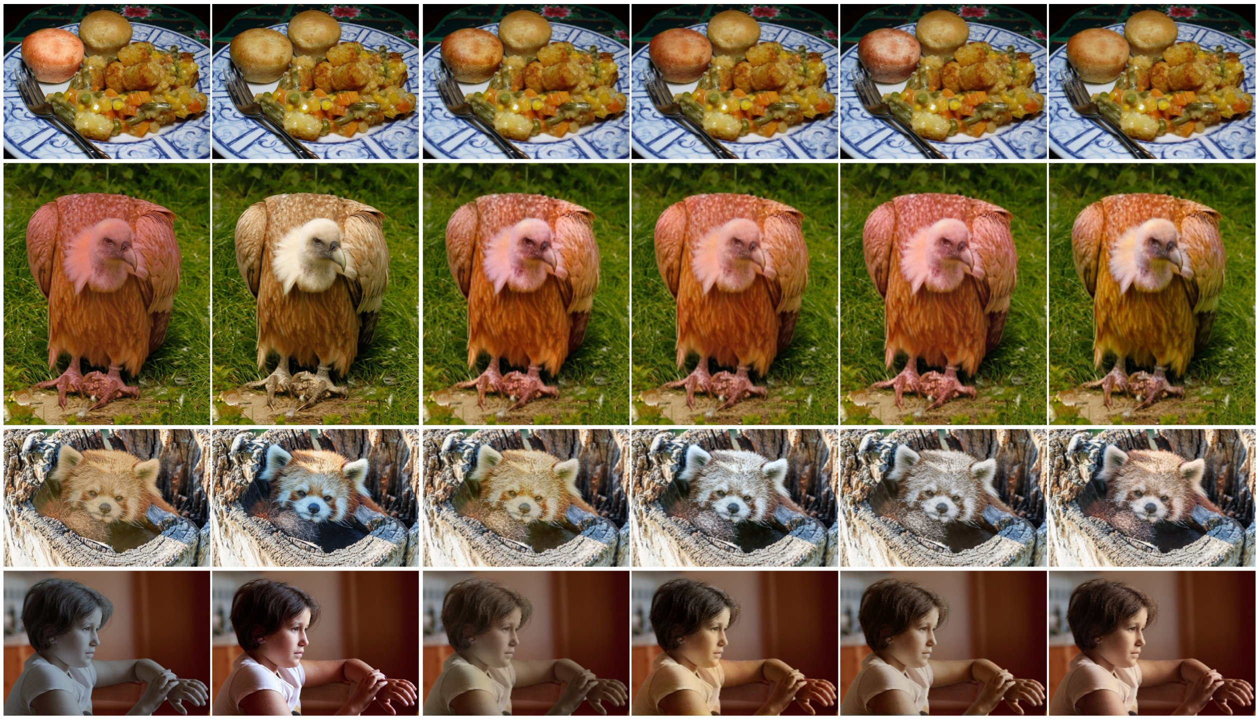

Figure 5: Qualitative comparison of different models on samples from the iHarmony4 test set. Presented models are trained with horizontal flip and RRC augmentations, and FN-MSE objective function. More results in the supplementary material.

and can model a feature pyramid by its own structure. We aggregate the extracted features with the intermediate encoder feature map with concatenation.

Figure 4 shows a detailed diagram of our architecture based on HRNet+OCR [41]. We further refer to it as iDIH-HRNet, additionally specifying the backbone width. Illustrative results of previous works and our models are presented in Fig. 5.

3.4. Foreground-normalized MSE

As we mentioned in section 3.1, the characteristic of image harmonization task is that the background region of the output image should remain unchanged relatively to the input composite. When the network takes the foreground mask as an input, it learns to copy the background region easily. Hence, the pixel-wise errors in this region will become close to zero during training. This means that training samples with foreground objects of different sizes will be trained with different loss magnitudes, which leads to poor training on images with small objects. To address this issue, we propose a modification of the MSE objective function, which is normalized by the foreground object area:

$$\mathcal{L}_{rec}(\hat{I}) = \frac{\sum\limits_{h,w} \left\| I^{pred}_{h,w} - I_{h,w} \right\|_2^2}{\max\left\{ A_{min}, \sum\limits_{h,w} M_{h,w} \right\}}, \tag{2}$$

where A_{min} is a hyperparameter that prevents instability of the loss function on images with very small objects. This objective function becomes equal to simple MSE when $A_{min} = HW$. In all our experiments, we set $A_{min} = 100$, you can find the ablation studies on it in Section 4.4.

4. Experiments

4.1. Datasets

We use iHarmony4 dataset contributed by [5]. Images in the dataset contain a wide range of the foreground objects, and each object occupies a sufficient area of the image. Details on four iHarmony4 subdatasets are given below. Each subdataset is split into training and test sets so that each image is included in either a training or test set.

HCOCO. COCO dataset [23] contains 118k training and 41k test images with instance masks for 80 categories annotated manually. HCOCO subdataset is synthesized from the joined training and test sets of COCO. The composite im-

Method	HCOCO		HAdobe5k		HFlickr		Hday2night		All	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
DIH [34, 5]	51.85	34.69	92.65	32.28	163.38	29.55	82.34	34.62	76.77	33.41
S^2AM [6, 5]	41.07	35.47	63.40	33.77	143.45	30.03	76.61	34.50	59.67	34.35
DoveNet [5]	36.72	35.83	52.32	34.34	133.14	30.21	54.05	35.18	52.36	34.75
DIH+augs+FN-MSE	22.54	37.77	35.58	35.30	90.53	32.00	64.13	36.37	34.70	36.38
iDIH+augs+FN-MSE	19.29	38.44	30.87	36.09	84.10	32.61	55.24	37.26	30.56	37.08
iDIH-HRNet-W18s	**14.01**	**39.64**	**21.36**	**37.35**	**60.41**	**34.03**	**50.61**	**37.68**	**22.00**	**38.31**

Table 1: Performance comparison between methods on the iHarmony4 test sets. The best results are in bold.

Foreground ratios	$0\% \sim 5\%$		$5\% \sim 15\%$		$15\% \sim 100\%$		$0\% \sim 100\%$	
	MSE↓	fMSE↓	MSE↓	fMSE↓	MSE↓	fMSE↓	MSE↓	fMSE↓
DIH [34, 5]	18.92	799.17	64.23	725.86	228.86	768.89	76.77	773.18
S^2AM [6, 5]	15.09	623.11	48.33	540.54	177.62	592.83	59.67	594.67
DoveNet [5]	14.03	591.88	44.90	504.42	152.07	505.82	52.36	549.96
DIH+augs+FN-MSE	9.85	420.90	29.39	329.04	100.05	324.41	34.70	375.59
iDIH+augs+FN-MSE	8.38	366.32	25.39	287.02	89.44	297.94	30.56	330.45
iDIH-HRNet-W18s	**6.73**	**294.76**	**18.03**	**204.69**	**63.02**	**207.82**	**22.00**	**252.00**

Table 2: MSE and foreground MSE (fMSE) of different methods in each foreground ratio range based on the whole iHarmony4 test set. The best results are in bold.

age in HCOCO consists of the background and foreground areas both drawn from the real image, although the foreground appearance is modified. The foreground color information is transferred from another image foreground belonging to the same category. HCOCO subdataset contains 38545 training and 4283 test pairs of composite and real images.

HFlickr. Flickr is a website for online photo management and sharing. HFlickr is based on 4833 images crawled from Flickr with one or two manually segmented foreground objects. The process of the composite image construction is the same as for HCOCO. During compositing, a pre-trained scene-parsing ADE20K model [46] is used to specify the category of the foreground object. HFlickr subdataset contains 7449 training and 828 test pairs of composite and real images.

HAdobe5k. MIT-Adobe FiveK dataset [2] is collected from 5k raw photos, each having 5 renditions made by 5 different experts. 4329 images with one manually segmented foreground object are used to build the HAdobe5k subdataset. The background of the composite image is retrieved from the raw photo and the foreground is retrieved from one of the 5 edited images. HAdobe5k subdataset contains 19437 training and 2160 test pairs of composite and real images.

Hday2night. Day2night dataset [47] contains 8571 images of 101 outdoor scenes. It is collected from the AMOS dataset [13], which contains a large number of images taken every half an hour by outdoor webcams with fixed characteristics. 106 target images from 80 scenes with one manually segmented foreground object are selected to synthesize the Hday2night subdataset. The process of the composite image construction is the same as for HAdobe5k. The background and foreground images are taken from different pictures of one scene. Hday2night subdataset contains 311 training and 133 test pairs of composite and real images.

4.2. Training details

We use the PyTorch framework to implement our models. The models are trained for 180 epochs with Adam optimizer [16] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 10^{-8}$. Learning rate is initialized with 0.001 and reduced by a factor of 10 at epochs 160 and 175. The semantic segmentation backbone weights are initialized with weights from the models pre-trained on ImageNet and updated at every step with the learning rate multiplied by 0.1. All models are trained on the iHarmony4 training set, which encapsulates training sets from all four subdatasets.

We resize input images as 256×256 during both training and testing. The input images are scaled to $[0, 1]$ and normalized with RGB mean $(0.485, 0.456, 0.406)$ and standard deviation $(0.229, 0.224, 0.225)$. Training samples are augmented with the horizontal flip and random size crop with the size of the cropped region not smaller than the halved

Model	HFlip	RRC	FN-MSE	MSE↓	fMSE↓	PSNR↑
DIH	–	–	–	65.65	632.86	34.13
DIH	+	–	–	55.44	555.92	34.73
DIH	+	+	–	36.97	398.26	35.92
DIH	+	+	+	34.70	375.59	36.38
iDIH	+	+	–	33.96	378.25	36.53
iDIH	+	+	+	30.56	330.45	37.08
iS^{2}AM	+	+	–	30.46	335.98	36.91
iS^{2}AM	+	+	+	26.92	286.67	37.67

Table 3: Ablation studies of different augmentation strategies and the proposed FN-MSE objective function."HFlip" stands for the horizontal flip and "RRC" stands for the RandomResizedCrop augmentation. The iDIH and iS^{2}AM models include the layer blending original image with decoder output.

A_{min}	1		10		100		1000		10000	
	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR	MSE	PSNR
DIH	124.79	31.87	42.36	35.52	34.70	<u>36.38</u>	**33.84**	**36.43**	<u>34.24</u>	36.33
iDIH	32.41	36.87	**30.50**	**37.08**	<u>30.56</u>	<u>37.08</u>	31.19	37.03	31.93	36.81

Table 4: Ablation studies of the hyperparameter A_{min} value in the FN-MSE loss. The best results for each model are in bold, second to the best results are underlined.

input size. The cropped image is resized to 256×256 then.

4.3. Comparison with existing methods

The detailed comparison between traditional and deep learning methods is demonstrated in previous works [34, 5], showing that deep learning approaches generally perform better, so we compare our method with them only. We implement two baseline methods, S^{2}AM [6] and DIH [34] without the segmentation branch. In the iDIH model a final image is obtained by blending an original image and a decoder output using a predicted blending mask (see 3.1).

We provide MSE and PSNR metrics on the test sets for each subdataset separately and for combination of datasets in Table 1. Following [5], we study the impact of the foreground ratio on the model performance. To do that, we introduce the foreground MSE (fMSE) metric which computes MSE for the foreground area only. The metrics on the whole test set across three ranges of foreground ratios are provided in Table 2. The results show that our final model outperforms the baselines not only on the whole test set but also on each foreground ratio.

4.4. Ablation studies

We propose the modifications of the baseline method that generally improve harmonization. First, we upgrade training process for the DIH model by adding the horizontal flip and random size crop augmentations to increase the diversity of the training data. The results show that data augmentation is crucial for the harmonization model training and leads to significant metrics improvements. Second, the additional blending layer preserving the background details is then added to DIH resulting in iDIH. With these adjustments, the DIH architecture performs significantly better than the original one, so they are preserved in all further experiments. Then we compare MSE and FN-MSE objective functions on the DIH and iDIH models. The use of FN-MSE shows a consistent improvement in metrics. The detailed results of these ablation studies are shown in Table 3.

The iDIH structure is much lighter than S^{2}AM, which training takes 104 hours on a single GTX 2080 Ti, while the iDIH training requires just 13 hours. Considering that S^{2}AM requires much longer training time and tends to get metric values inferior to iDIH, we proceed with the latter.

FN-MSE function depends on the A_{min} hyperparameter, so we conduct experiments on its choice, which are presented in Table 4. The DIH model achieves the best results when the hyperparameter is set to $A_{min} = 10$, while the iDIH model performs better with $A_{min} = 1000$. Models tend to be unstable when A_{min} is too small due to possible high loss values on tiny objects. As you can notice, the DIH and iDIH models show nearly the best results with $A_{min} = 100$. We decided to use that value in all our experiments as the most stable one.

We examine the impact of different backbones on the proposed architecture. We train the model with three different HRNet models: HRNetV2-W18 and HRNetV2-W32 with 4 high-resolution blocks, HRNetV2-W18s with 2 high-resolution blocks. We observe that increasing HRNet ca-

Model	MSE↓	fMSE↓	PSNR↑
iDIH-HRNet-W18s	**22.00**	**252.00**	**38.31**
iDIH-HRNet-W18s non foreground-aware HRNet	26.95	301.39	37.56
iDIH-HRNet-W18s pass features using feature pyramid	22.39	256.35	38.23
iDIH-HRNet-W18	22.55	255.93	38.31
iDIH-HRNet-W32	23.10	264.58	38.22
iDIH-DeepLabV3+ (R-34)	27.79	310.14	37.49
iS^{2}AM-HRNet-W18s	25.14	273.14	37.93
iS^{2}AM-HRNet-W18	23.22	254.18	38.26
iS^{2}AM-HRNet-W32	24.09	266.27	38.00

Table 5: Ablation studies of the backbone model choice. In all models the backbone features are passed to both encoder and decoder simultaneously.

Encoder	Decoder	Pre-trained HRNet	MSE↓	fMSE↓	PSNR↑
–	+	–	23.08	263.41	38.07
–	+	+	23.39	261.30	38.18
+	–	–	23.74	265.55	38.01
+	–	+	22.85	259.77	38.18
+	+	–	22.42	256.04	38.17
+	+	+	**22.00**	**252.00**	**38.31**

Table 6: Ablation studies of different backbone aggregation strategies in iDIH-HRNet-W18s. "Encoder" and "Decoder" columns denote the stages at which the backbone features are passed to the model.

pacity does not improve quantitative results significantly. In addition, we conduct an experiment with ResNet-34 & DeepLabV3+ [3] and do not notice any improvements regarding HRNet. The results of models with different backbones are shown in Table 5.

One can notice that only one extending the iDIH architecture with HRNet makes the final model more advanced and increases its capacity. The question arises whether pre-training on ImageNet improves results or not. To answer this question we conduct several experiments without initializing HRNet with pre-trained weights. We observe that models initialized with pre-trained weights show constantly better results, as can be seen in Table 6. Nevertheless, even the models trained from scratch show much better metrics than pure iDIH. This means that the proposed architecture itself works very well on the image harmonization task and pre-training helps to improve metrics further.

5. Conclusion

We propose a novel approach to incorporating high-level semantic features from models pre-trained on the ImageNet dataset into the encoder-decoder architectures for image harmonization. We also present a new FN-MSE objective function proven to be effective for this task. Furthermore, we observe that the use of simple training augmentations significantly improves the performance of the baselines. Experimental results show that our method is considerably superior to existing approaches in terms of MSE and PSNR metrics.

References

[1] Rodrigo Benenson, Stefan Popov, and Vittorio Ferrari. Large-scale interactive object segmentation with human annotators. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 11700–11709, 2019.

[2] Vladimir Bychkovsky, Sylvain Paris, Eric Chan, and Frédo Durand. Learning photographic global tonal adjustment with a database of input/output image pairs. In *CVPR 2011*, pages 97–104. IEEE, 2011.

[3] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

[4] Daniel Cohen-Or, Olga Sorkine, Ran Gal, Tommer Leyvand, and Ying-Qing Xu. Color harmonization. In *ACM SIGGRAPH 2006 Papers*, pages 624–630. 2006.

[5] Wenyan Cong, Jianfu Zhang, Li Niu, Liu Liu, Zhixin Ling, Weiyuan Li, and Liqing Zhang. Dovenet: Deep image harmonization via domain verification. *arXiv: Computer Vision and Pattern Recognition*, 2020.

[6] Xiaodong Cun and Chi-Man Pun. Improving the harmony of the composite image by spatial-separated attention module. *IEEE Transactions on Image Processing*, 29:4759–4771, 2020.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[8] Caner Hazirbas, Lingni Ma, Csaba Domokos, and Daniel Cremers. Fusenet: Incorporating depth into semantic segmentation via fusion-based cnn architecture. In *Asian conference on computer vision*, pages 213–228. Springer, 2016.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[10] Mingming He, Dongdong Chen, Jing Liao, Pedro V Sander, and Lu Yuan. Deep exemplar-based colorization. *ACM Transactions on Graphics (TOG)*, 37(4):1–16, 2018.

[11] Tobias Hinz, Matthew Fisher, Oliver Wang, and Stefan Wermter. Improved techniques for training single-image gans. *arXiv preprint arXiv:2003.11512*, 2020.

[12] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017.

[13] Nathan Jacobs, Nathaniel Roman, and Robert Pless. Consistent temporal variations in many outdoor scenes. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–6. IEEE, 2007.

[14] Jiaya Jia, Jian Sun, Chi-Keung Tang, and Harry Shum. Drag-and-drop pasting. *ACM Trans. Graph.*, 25:631–637, 2006.

[15] Hyunsu Kim, Ho Young Jhoo, Eunhyeok Park, and Sungjoo Yoo. Tag2pix: Line art colorization using text tag with secat and changing loss. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 9056–9065, 2019.

[16] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[18] Jean-François Lalonde and Alexei A. Efros. Using color compatibility for assessing image realism. *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, 2007.

[19] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *European Conference on Computer Vision*, pages 577–593. Springer, 2016.

[20] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Colorization as a proxy task for visual understanding. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 840–849, 2017.

[21] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[22] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[23] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[24] Pengju Liu, Hongzhi Zhang, Kai Zhang, Liang Lin, and Wangmeng Zuo. Multi-level wavelet-cnn for image restoration. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 773–782, 2018.

[25] Wenjie Luo, Yujia Li, Raquel Urtasun, and Richard Zemel. Understanding the effective receptive field in deep convolutional neural networks. In *Advances in neural information processing systems*, pages 4898–4906, 2016.

[26] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. Image colorization using generative adversarial networks. In *International conference on articulated motion and deformable objects*, pages 85–94. Springer, 2018.

[27] Patrick Pérez, Michel Gangnet, and Andrew Blake. Poisson image editing. In *ACM SIGGRAPH 2003 Papers*, pages 313–318. 2003.

[28] François Pitié and Anil Kokaram. The linear monge-kantorovitch linear colour mapping for example-based colour transfer. 2007.

[29] Erik Reinhard, Michael Adhikhmin, Bruce Gooch, and Peter Shirley. Color transfer between images. *IEEE Computer graphics and applications*, 21(5):34–41, 2001.

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[31] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. Singan: Learning a generative model from a single natural image. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4570–4580, 2019.

[32] Jian Sun, Jiaya Jia, Chi-Keung Tang, and Heung-Yeung Shum. Poisson matting. In *ACM SIGGRAPH 2004 Papers*, pages 315–321. 2004.

[33] Kalyan Sunkavalli, Micah K. Johnson, Wojciech Matusik, and Hanspeter Pfister. Multi-scale image harmonization. In *SIGGRAPH 2010*, 2010.

[34] Yi-Hsuan Tsai, Xiaohui Shen, Zhe L. Lin, Kalyan Sunkavalli, Xin Lu, and Ming-Hsuan Yang. Deep image harmonization. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2799–2807, 2017.

[35] Jue Wang, Maneesh Agrawala, and Michael F Cohen. Soft scissors: an interactive tool for realtime high quality matting. In *ACM SIGGRAPH 2007 papers*, pages 9–es. 2007.

[36] Jingdong Wang, Ke Sun, Tianheng Cheng, Borui Jiang, Chaorui Deng, Yang Zhao, Dong Liu, Yadong Mu, Mingkui Tan, Xinggang Wang, Wenyu Liu, and Bin Xiao. Deep high-resolution representation learning for visual recognition. *TPAMI*, 2019.

[37] Yifan Wang, Federico Perazzi, Brian McWilliams, Alexander Sorkine-Hornung, Olga Sorkine-Hornung, and Christopher Schroers. A fully progressive approach to single-image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 864–873, 2018.

[38] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Gp-gan: Towards realistic high-resolution image blending. In *Proceedings of the 27th ACM International Conference on Multimedia*, pages 2487–2495, 2019.

[39] Ning Xu, Brian Price, Scott Cohen, Jimei Yang, and Thomas S Huang. Deep interactive object selection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 373–381, 2016.

[40] Su Xue, Aseem Agarwala, Julie Dorsey, and Holly Rushmeier. Understanding and improving the realism of image composites. *ACM Trans. Graph.*, 31:84:1–84:10, 2012.

[41] Yuhui Yuan, Xilin Chen, and Jingdong Wang. Object-contextual representations for semantic segmentation. *arXiv preprint arXiv:1909.11065*, 2019.

[42] Lingzhi Zhang, Tarmily Wen, and Jianbo Shi. Deep image blending. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 231–240, 2020.

[43] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8827–8836, 2018.

[44] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016.

[45] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 286–301, 2018.

[46] Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019.

[47] Hao Zhou, Torsten Sattler, and David W Jacobs. Evaluating local features for day-night matching. In *European Conference on Computer Vision*, pages 724–736. Springer, 2016.

[48] Jun-Yan Zhu, Philipp Krahenbuhl, Eli Shechtman, and Alexei A Efros. Learning a discriminative model for the perception of realism in composite images. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3943–3951, 2015.