Learning to Blend Photos

Wei-Chih Hung[1][*], Jianming Zhang[2], Xiaohui Shen[3][†], Zhe Lin[2],
Joon-Young Lee[2], and Ming-Hsuan Yang[1,4]

[1]UC Merced   [2]Adobe Research   [3] Bytedance Lab   [4]Google Cloud

Fig. 1. Sample blending photos. These results are generated by our proposed system to simulate the "Double Exposure" effect with different blending engines. Best viewed in color.

Abstract. Photo blending is a common technique to create aesthetically pleasing artworks by combining multiple photos. However, the process of photo blending is usually time-consuming, and care must be taken in the process of blending, filtering, positioning, and masking each of the source photos. To make photo blending accessible to general public, we propose an efficient approach for automatic photo blending via deep learning. Specifically, given a foreground image and a background image, our proposed method automatically generates a set of blending photos with scores that indicate the aesthetics quality with the proposed quality network and policy network. Experimental results show that the proposed approach can effectively generate high quality blending photos with efficiency.

---

[*]This work was done during the author's internship at Adobe Research.
[†]This work was done when the author was at Adobe Research.

1 Introduction

Photo blending is a common technique in photography to create aesthetically pleasing artworks by superimposing multiple photos together. For example, "Double Exposure" is one of the popular effects in photo blending (see Figure 1), which can be achieved with skillful camera capture techniques or advanced image editing software. This effect is widely used in posters, magazines, and various print advertisements to produce impressive visual effects and facilitate story-telling. However, the process of photo blending is usually time-consuming even for experts and requires a lot of expertise. To make this process faster, many image editing software products like Photoshop support the use of action scripts to simplify the photo blending operations. But these scripts are predefined and do not consider how to adjust the photos based on the context. Therefore, they may not work out-of-the-box, and it still takes a fairly amount of skill and time to tweak the results to the users' satisfaction.

By observing blending artworks, we identify two critical factors in the making of a satisfactory double exposure effect: **background alignment** and **photometric adjustment**. Since the background photo can contain scene elements of different textures and colors, how to align the background elements with the foreground object is of great importance for appealing blending results. On the other hand, since the blending function is a numerical function of two pixel values from foreground and background, the photometric adjustment of foreground photo, including brightness, contrast, and color modification, can affect the visibility of different regions in the photos, leading to visually different results.

In this work, we propose a fully automatic method to generate appealing double exposure effects by jointly predicting the background alignment region and the photometric adjustment parameters. The first challenge is how to design an evaluation metric to access the aesthetics quality of the blending results. Though many works [18,21,30,31,39] have been proposed for general photo quality assessment, we find that the result rankings of these methods are not consistent with user preference since the models are trained with common photos. Furthermore, these methods usually train a Convolutional Neural Network (CNN) to directly predict the provided fine-level aesthetics score from existing datasets [34,21,39]. However, we find that it is difficult for users to annotate the fine-level scores on these artworks such as photo blending. Therefore, we train a quality network on a newly collected dataset with course level annotation. We also propose to use a combination of ranking loss and binary cross entropy loss for as we find that it improves the training stability.

Given the proposed quality network, the second task is to find a region of interest (ROI) in the given background photo and a set of photometric adjustment parameters for the foreground photo, generating a blending result with the optimal rating according to the quality network. We view this problem as a derivative-free optimization task, which is subject to parameter search range and time constraints. There exist many non-learning based search methods for global function optimization, e.g., grid search, random search, simulated annealing [20], or particle swarm optimization [19]. However, in our experiments, we find that

these methods could not find a good optimum within the constrained time since the system has to blend the images for each set of selected parameters and pass it through the quality network. Therefore, we propose to train an agent with deep reinforcement learning (DRL) to search for the best combination of background ROI and foreground adjustment. Specifically, we transform the quality network to a two-stream policy network that outputs both state quality value and action values to make the search process efficient. To evaluate the proposed algorithm, we compare the proposed method with existing search algorithms, e.g., Particle Swarm Optimization [19] and Simulated Annealing [20], showing that the proposed DRL search generates results with higher quality values with the same time constraint. Also, we conduct the user study to compare our method with other baselines as well as human experts. The results show that the quality scores indicated by the proposed quality network are consistent with user preference, and the proposed method effectively generates aesthetically pleasing blending photos within few seconds.

The contributions of this work are as follows. First, to the best of our knowledge, it is the first work to introduce the task of automatic photo blending. Second, we propose a quality network trained with ranking loss and binary cross entropy loss using a newly collected dataset for double exposure. Third, we transform the quality network into a two-stream policy network trained with deep reinforcement learning, performing the automatic photo blending with consideration of user preference, aesthetics quality, image contexts, as well as a tight runtime constraint.

2   Related Work

Learning-based photo editing. Recently, many CNN based methods for image editing tasks have been proposed with impressive results, such as image filtering [29,24,22], enhancement [1,14,50,9], inpainting [38,51], composition [57,45,44,49], colorization [15,23,54,56], and image translation [16,58,3,28,59]. Most of these approaches consist of a CNN model that directly transforms a single input photo to the desired output. However, since there is no specific constraint on how the CNNs transform the photos, visual artifacts are inevitable in some of the results. Also, the resulting photos' resolution is usually low because of the limited GPU memory. Though Gharbi *et al.* [9] propose to use deep bilateral learning to process high-resolution images, it only works with the effects that can be interpreted as pixel affine transform. In this work, since all the image processing modules are predefined such as pixel blending, photometric adjustment, or predefined filtering, the blending results generated by our method is artifact-free, and there is no limited resolution since most image processing modules operate with CPU. Style transfer methods [40,8,6,13,17,25,26,27,32] also relate to our task. In style transfer, an input photo is stylized with a reference style image by matching the correlations between deep features while preserving the content with the perceptual loss on higher level features. In our task, content preservation is also important. While style transfer methods preserve the con-
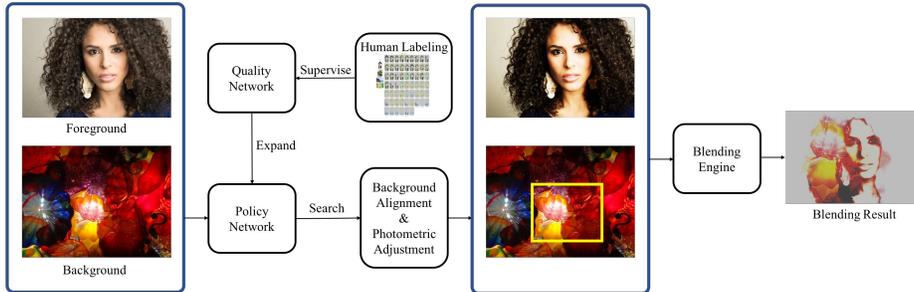
Fig. 2. Overview of the automatic blending framework. The inputs of our method are two photos: foreground and background. We first train a quality network to evaluate the aesthetics quality of blending photos with human preference annotation on random blending photos. Then a deep reinforcement learning based agent is trained to optimize the parameter for the background alignment and photometric adjustment. Using the predicted parameters, the blending engine renders the final blending photo.

tent with perceptual loss [17], we find that the perceptual loss values are not consistent with the blending results, i.e., the loss is not always low when the blending results preserve the content well.

Aesthetics quality assessment. The objective of aesthetics quality assessment methods is to rank the photos according to the general aesthetics preference automatically. These methods can be applied to image search, album curation, and composition ranking. Recently, many methods [18,21,30,31,33,39] are proposed based on CNN models. However, when we test these models with the blending photos, we find that the result rankings are not consistent with user preference since the models are trained with common photos.

Deep reinforcement learning. Recently, deep reinforcement learning methods [36,48,46,35] have draw much attention due to the great success in computer games [36] and board games [41]. Our approach of learning a quality network from user annotation is closely related to some existing reward learning methods such as inverse RL [37,7], imitation learning [12,43], and human-in-the-loop RL [5], where the reward function is not explicitly available in the environment. In vision community, some works have been proposed to apply DRL for object localization [2], visual tracking [53], and object detection [52]. Similar to our proposed background alignment approach, these methods model the object localization problem as a sequence of actions that move or scale the ROIs. The major difference between our work and these methods is that we are not searching for a single object but a suitable blending position that considers both the foreground and background context, as well as the subjective user preference.

3   System Overview

Our goal is to develop a method that automatically generates the aesthetically pleasing blending results based on the given foreground and background image pairs. Figure 2 shows the framework of the proposed system. In our proposed method, a DRL-based agent searches for the optimal parameters of background alignment and photometric adjustment based on the input context and a selected blending engine, which renders the final blending results with the parameters selected by the agent. To train the agent, an evaluation function is necessary to generate the quality metric, i.e., to tell how good the current blending result is during the optimization process. In typical RL environments, the reward is usually well defined, e.g., the scores (win or lose) in games environment. However, there is no well-defined evaluation function for artworks such as photo blending. Therefore, we propose to learn an evaluation from user annotation. Specifically, we generate blending results with random parameters on the collected foreground and background images. We then invite participants to evaluate the blending results based on their subjective preference. Based on the labels, we train a quality network served as the evaluation function for blending results. Once we have the evaluation function, we train the DRL agent to search in the parameter space effectively with the proposed policy network and existing DRL algorithms.

4   Quality Network

Our objective is to learn a function that assesses a blending photo with a numerical score that indicates the aesthetic quality, and the quality scores should be consistent with the user preferences, i.e., the higher user rating suggests higher quality score, and vice versa. We observe that most people evaluate the blending results by comparing them with the original foreground photos. If one blending photo does not preserve the original image contexts well, the users would often rate it as unacceptable. However, if the blending photo preserves context but fails to have artistic effects, it would still not be rated as a good one. Therefore, we consider the evaluation function conditional on the foreground image and build it with a CNN model that takes both the blending photo and the foreground image as input. We denote the proposed CNN as quality network since it indicates the aesthetic quality of the blending result.

4.1   Network Structure

Figure 3 shows the structure of the proposed quality network. The network is composed of two VGG16 networks [42] pre-trained on ImageNet as feature extractors where the weights are shared as a Siamese network [4]. We remove the last classifier layer of VGG and concatenate the features of both base networks, and add two fully-connected layers with the 512-channel as the middle layer to output a single score. It takes the foreground image and a blended image as input, which both are downsized to 224×224, and outputs a single scalar as the blending reward score.
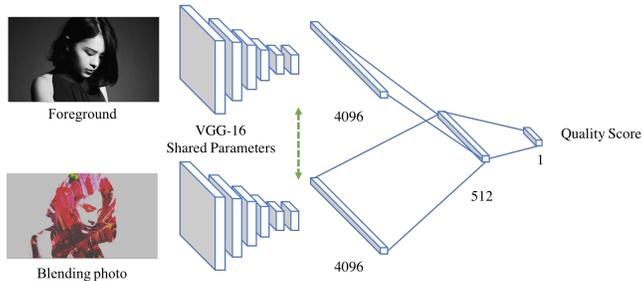
Fig. 3. Network structure of the proposed quality network. The quality network consists of two of VGG16 networks that share the same parameters. The quality network takes the original foreground photo and the blending results as input and outputs a numeric score that indicates the aesthetic quality of the blending result.

4.2   Learning objective

Given two blending images B_i and B_j as well as their original foreground image F_i and F_j, our objective is to train a mapping function $S(\cdot)$ (quality network) that maps (F_i, B_i) and (F_j, B_j) to two numerical scores, such that if B_i is more visually pleasing than B_j according to user rating then

$$S(F_i, B_i) > S(F_j, B_j), \tag{1}$$

and vice versa. To achieve this, we use the ranking loss $\mathcal{L}_r$ as the training loss function for the quality network. We denote $S_i = S(F_i, B_i)$, and let $y = 1$ if S_i has better user rating than S_j, while $y = -1$ otherwise. We formulate the loss as

$$\mathcal{L}_r(S_i, S_j) = \max\{0, -y(S_i - S_j) + m\}, \tag{2}$$

where m is the margin term.

However, the ranking loss only enforces the property that for a given photo the scores of good examples are higher than the fair/bad ones but lack of a universal threshold of differentiating good ones for every input photo. If the score ranges are consistent with different input sets, one can choose a score threshold to filter out the blending results that are not acceptable by most users. Therefore, we propose to add additional binary cross entropy loss on top of the predicted scores to enforce that all the bad examples have scores that are less than zero. The binary cross entropy loss function can be formulated as

$$\mathcal{L}_{bce}(S_i) = -r \log(\sigma(S_i)) - (1 - r) \log(1 - \sigma(S_i)), \tag{3}$$

where $r = 1$ if the user labeling is "good/fair", and $r = 0$ if the user labeling is "bad", and $\sigma(\cdot)$ is the sigmoid function. Combining the ranking loss and cross entropy loss, the overall optimization objective then becomes $\mathcal{L} = \mathcal{L}_r + \lambda\mathcal{L}_{bce}$, where both the ranking property and the score offset can be preserved. Please see supplementary materials for implementation details.

foreground   background   -16.57   -2.01   5.38   12.65
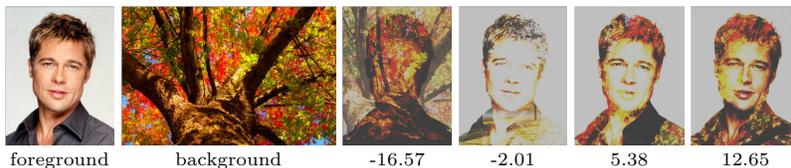
Fig. 4. Sample blending results with quality scores computed by the quality network. A higher score suggests that the blending result has better aesthetics quality, and a lower score suggest that most users do not find the result appealing.

Figure 4 shows some example blending results with their scores indicated by our trained quality network. Among the results of high scores, the background usually has good alignment with foreground, and the brightness/contrast is adequate to control the level of blending.

4.3 Dataset collection

To generate the blending images, we download 5,000 portrait photos from the internet as foreground images, as well as 8,000 scenic photos as background images. Then we apply the blending engines on random pairs of foreground and background images with random alignment/adjustment parameters. During the labeling process, the users are asked to label each blended image a score according to their preference. In our implementation, the preference score has three levels: "good", "fair", and "bad", where we provide some basic guidelines: "good" denotes the one that one likes, "bad" represents the one would like to discard, and "fair" means the one is acceptable but needs to be further adjusted to make it better.

However, we find that most of the randomly generated blending results are of worse quality, where the original foreground context (face) is often not recognizable and will be annotated as "Bad" for almost all annotators. To increase the labeling efficiency, we first train a quality network as described in Section 4 with 5,000 ratings from annotators, who are asked only to consider how well can you recognize the original foreground content. Then we apply the quality network to all the generated blending results and filter out the results that have scores below a designed threshold. As a result, we collect 30,000 ratings on 1,305 image sets with 16 annotators.

5 Deep Reinforcement Search

Given the quality network, we seek to predict a region of interest (ROI) on the background image and the photometric adjustment parameters that could generate the highest score concerning the quality network. We view the problem as a derivative-free optimization task, which is subject to parameter search range and time constraints. There exist many non-learning based search methods for
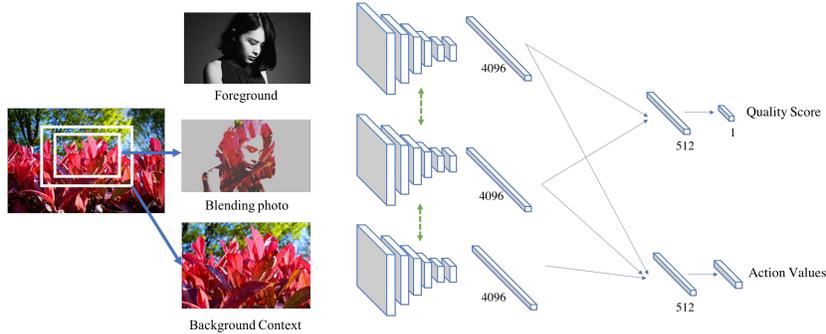
Fig. 5. Network structure of the proposed policy network. We extend the quality network with an additional background context input and action values output. The designed structure enables the agent to jointly predict the quality value as well as search action values, reducing the time complexity during the process of parameter search.

global function optimization, e.g., grid search, random search, simulated annealing [20], or particle swarm optimization [19]. However, in our experiments, we find that these methods could not find a good optimum within the constrained time since the system has to blend the images for each set of selected parameters and pass it through the quality network.

To tackle the above-mentioned problem, we introduce a DRL-based agent to predict actions for searching the parameters efficiently. Given the selected foreground image, background image, and blending engine, the agent takes the state values and pass them through the proposed policy network to get the action values. The agent then performs the action with the highest action value and getting an immediate reward that suggests the blending quality changes. The goal of the agent is to learn a policy for maximizing the expected accumulated future award. In our task, it is equivalent to maximizing the score of the final blending result output by the quality network.

5.1   Search space and actions

We define total ten actions for the DRL agent, in which six actions are used to move the current ROI for alignment: (Right, Left, Up, Down, Bigger, Smaller), and the other four actions are for the foreground photometric adjustment: (Brightness+, Brightness-, Contrast+, Contrast-). All the actions are performed relatively. For instance, the action "Right" means moving the ROI right for $\alpha \times w$, where w is the width of current ROI, and α is set to 0.05 in our experiments. It is similar for photometric actions, where "Brightness+" means increasing the pixel values for a ratio β, which is set to 0.1. We provide the detail action operations in the supplementary materials.

Foreground Agent steps   Initial   Final

Fig. 6. An example of the intermediate DRL behavior. We show the ROI actions with color encoding. The initial ROI is orange while the last ROI is green. During the process, the brightness is decreased to 0.9, and the contrast is increased to 1.2.

5.2   Network structure

We show the structure of the policy network in Figure 5. In deep reinforcement learning, a policy network takes the observation states as input and output the action values which indicates the expected accumulated reward after performing corresponding actions. In the proposed method, we select three state variables: foreground image, the blended image concerning the currently selected region, and the background context. The background context is the enlarged background region that enables the policy network to see what are the potential directions to go for optimizing the expected reward. In our experiments, we choose the context ROI as 1.3 times larger than the blending ROI. Based on the pretrained quality network in Figure 3, we add an input stream for background context information as well as the action values output. Since the designed structure enables the agent to jointly predict the quality value as well as search action values, we could record the quality score output of policy network during the test time while performing the actions and use the state with the maximum score as the final result.

5.3   Reward shaping

We set the reward value as the score difference after performing the selected action:

$$R_t = S(F, B_t) - S(F, B_{t-1}), \qquad (4)$$

where R_t is the reward value at step t, $S(\cdot)$ is the quality network, F is the foreground image, and B_t is the blending results at step t. That is, if the selected action increases the score, we provide a positive reward to encourage such behavior. Otherwise, we provide a negative reward to discourage it.

5.4   Implementation Details

We train the DRL agent Dueling-DQN [48] as well as A2C [47], which is a synchronous version of A3C [35] that enables GPU training. Training a model

requires around 20 hours to get stable performance for both methods. The details of the training process and parameters can be found in the supplementary materials. During the training process, the agent randomly picks a foreground image, a background image, and a random initial ROI for simulation. We set the maximum steps as 50 for each episode since there is no terminal signal in our task. When the selected action causes ROI to outside the background image or causing extreme photometric adjustment value, we provide a negative reward of value -1.0 to the agent and maintain the current state. We show an example of the intermediate search steps in Figure 6. The code and data will be made available at https://github.com/hfslyc/LearnToBlend.

6   Blending Engine

The blending process is deterministic and composed of three major components: pre-processing, pixel blending, and post-processing. Pixel blending is a function that transforms two pixel values from foreground and background respectively to a blended pixel, and it is applied to every pixel pair after background is cropped and aligned with the foreground. The commonly used pixel blending functions are often simple mathematical functions, e.g., Add, Subtract, or Multiply. In this work, we focus on the most widely used blending variant: Double Exposure, in which the blending mode is called "Screen". The blending function can be formulated as:

$$x_i^{blend} = 1 - (1 - x_i^{fg}) \cdot (1 - x_i^{bg}), \tag{5}$$

where x_i^{fg}, x_i^{bg} are two pixels of location i from the foreground and cropped background images, and x_i^{blend} is the "Screen" blended pixel values. We assume the pixel value range $[0, 1]$, and the function is applied to all color channels independently. According to (5), the resulting value would be bright (1.0) if either the foreground pixel or background pixel is close to 1.0. Since in most cases the foreground photos are brighter, the overall effect could be seen as the dark parts of the foreground replaced by the background as shown in Figure 1.

The pre-processing and post-processing could consist of any filtering, styling, or enhancement modules. For example, one engine can apply the Instagram filters and the background removal algorithms on foreground images either as pre- or post-processing modules. For simplicity, we carry out the experiments with only the foreground removal as pre-processing and one specific color tone adjustment as post-processing. We show more qualitative results with different styles of blending engine in the supplementary materials.

7   Experimental Results

7.1   Evaluation of quality network

To show the effectiveness of the proposed quality network, we compare the proposed method with Perceptual Loss [17] and Learned Perceptual Image Patch

Table 1. Evaluation of Quality network.

Method	Mean Accuracy
Perceptual Loss [17]	55.30%
LPIPS [55]	61.78%
Quality Network (no FG)	75.23%
Quality Network (no BCE)	76.17%
Quality Network	**79.69%**

Similarity (LPIPS) [55] between the foreground image and blending result since a higher perceptual similarity often implies better user rating. We collect a validation set with 3205 user ratings on 100 image sets as stated in Section 4.3. Among the ratings that correspond the same foreground image, we sample all possible good-bad and fair-bad pairs to evaluate the mean accuracy, i.e., blending results with better user ratings should have a higher quality score/perceptual similarity and lower perceptual loss. As shown in Table 1, the proposed quality network can align the user preference better than existing methods. We also carry out ablation studies to validate the design choices of the quality network in Table 1. When removing the foreground branch in the quality network (no FG), the accuracy drops by 4.46%, reflecting the fact that the result aesthetics quality often depends on the original foreground. In addition, without the binary cross entropy (no BCE), the accuracy drops 3.52%, showing that the binary cross entropy can effectively regularize the quality network.

Table 2. Comparisons of random search.

Table 3. Comparisons of search methods.

Method	Steps	Mean Q.
Random-10	10	5.69 ± 0.75
Random-50	50	7.08 ± 0.64
Random-100	100	8.07 ± 0.53
Random-500	500	9.47 ± 0.47
Random-1k	1,000	9.99 ± 0.41
Random-5k	5,000	11.51 ± 0.10
Random-10k	10,000	11.76 ± 0.06

Method	Steps	Mean Q.
Tree Search	150	7.15 ± 0.61
Gaussian Process	100	7.33 ± 0.35
Greedy	100	7.43 ± 0.66
Simulated Annealing	100	8.12 ± 0.63
PSO [19]	100	8.91 ± 0.44
Dueling-DQN	100	9.86 ± 0.30
A2C	100	10.93 ± 0.26

7.2 Evaluation of DRL search

In Table 2 and 3, we evaluate the effectiveness of the proposed DRL search. First, we select 20 input pairs as the evaluation set and perform the random search to obtain the upper bound of quality score that we can achieve. During the search, the ROIs and photometric parameters are randomly sampled within

the effective range to generate blending results, and the resulting photo is the one with the highest quality score. We report the search time cost in terms of evaluation steps since the forward time of quality network dominates the search process. We note that the forward time of the policy network is similar to the quality network. In our machine, each evaluation step takes 0.07 seconds with GPU. Note that during the search process, since only low-resolution results are rendered by blending, the overhead is much less than the final blending. As shown in Table 2, it will cost 10,000 evaluation steps, which takes around 11 minutes in our setup, to achieve the highest mean quality score of 11.76. However, in time-critical applications, complexity is of great importance. Therefore, we set a constraint to the evaluation steps as 100, and we compare the DRL based search with following derivative-free optimization methods:

- **Tree Search.** Use Random-50 as the initial point and search for all possible action sequences with depth 2.
- **Gaussian Process.** Use random 50 initial evaluation points and update the Gaussian approximation with 5 sampled points for 10 iterations. [22] also applies the Gaussian process for estimating the editing parameters.
- **Greedy.** Use Random-50 as the initial point and choose the best action for 5 steps (each cost 10 steps).
- **Simulated Annealing** [20]. (SA) Use Random-50 as the initial point and perform Simulated Annealing for 50 steps.
- **Particle Swarm Optimization** [19]. (PSO) Use 20 particles to perform 5 parallel updates.

All methods except PSO use Random-50 as the initial seed since it is a good trade-off between time cost and performance. We optimize the parameters of SA and PSO for best performance and report them in the supplementary materials. We show the comparisons in Table 3. Among the baselines, tree search performs the worst because it can only perform a depth-2 search within the time limit. Similarly, greedy search also suffers from the evaluation cost and can only perform 5 updates. SA can escape local optimum while performing the local search. But it only outperforms random-100 by a small margin, suggesting that the short schedule of SA does not converge well. Of all non-learning based methods, PSO performs the best with quality score 8.91 because of the joint combination of local and global optimization.

Both Dueling-DQN and A2C perform favorably against other baselines since the agent can perform different policy based on current image contexts for better exploration. A2C performs better than Dueling-DQN by 1.07, and we find that the non-deterministic action sampling (on-policy) of A2C helps to escape the local optimum, while with DQN the random exploration can only be used during training stage with ϵ-greedy.

7.3   User study

We set up a user study to evaluate the effectiveness of the proposed method. To compare with human experts, we ask an expert to generate the same effect with a predefined Photoshop action script that automatically performs Double
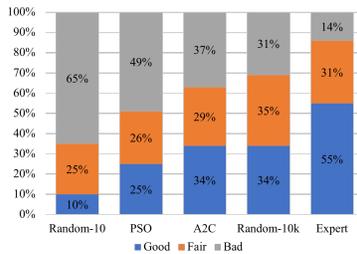
Fig. 7. Qualitative comparisons of selected baseline methods. We show the ratios of user ratings of selected baseline methods.

Exposure. To have a fair comparison with our baselines, the expert can only manipulate the background alignment and photometric adjustment. We record the expert adjusting process, and the average time for complete one blending photo is around 5 minutes with the help of action script. The user is asked to evaluate a total of 20 set images. During the study, each user sees five blending results for each foreground image that correspond to the following baselines: Random-10, PSO, Ours (A2C), Random-10k, and the human expert results. For each blending result, the user is asked to label each result as "good", "fair", or "bad". As a result, a total of 41 subjects participated in the study with a total of 4,100 votes.

We show the user study results in Figure 7 and show some qualitative comparisons in Figure 8. If we perform blending with only 10 random searches, 65% of blending results are considered bad to users, and only 10% of them are considered good. This shows that the task for photo blending is not trivial, as random parameters usually do not result in appealing results. Compared with Random-10, both PSO and Ours (A2C) obtain more aesthetically pleasing results. However, the proposed DRL search performs favorably against PSO with the same time cost (5 seconds) since it can exploit the current image contexts for better search policy. Random-10k represents the upper bound of the most aesthetically pleasing result that could be generated by the quality network. The performance of Random-10k only outperforms the proposed method by few percentage points but costs 7 more minutes per image, demonstrating that the proposed DRL search is an efficient way for searching in the parameter space.

The blending results of expert have the best aesthetics quality. The major difference is that the expert rarely makes blending results that are not acceptable to the user, resulting only 14% of "bad" ratings while our method has 37%. The fact that the exhaustive search (Random-10k) cannot outperform human expert with higher mean quality score (11.76 v.s. 11.32) suggests that there is still room for improving the proposed method.

Interestingly, we find that our method receives more "good" user ratings than the expert results in 6 sets out of 20 sets of blending results. It shows that our proposed method can, in some cases, produces results with higher quality than the one generated by the expert. However, there is no blending set where the

| Foreground | Random-10 | PSO | A2C | Expert |
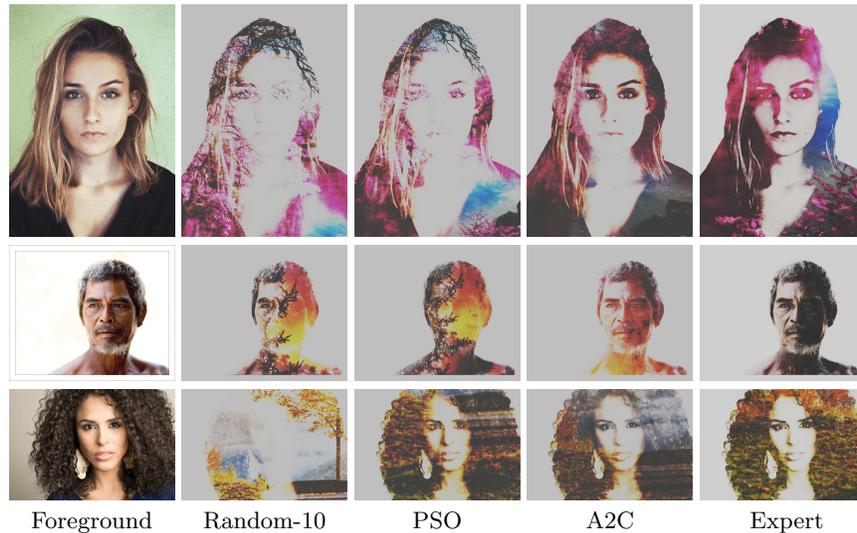
Fig. 8. Qualitative comparisons of different baseline methods. These results are generated with same foreground and background image pairs for each method. All of them are used in the user study.

expert result gets more "bad" ratings than other baselines. It suggests that even some people do not see the expert-generated results as the best one, they still do not consider them not acceptable.

The quality differences between the baseline methods are also consistent with the average quality scores indicated by the proposed quality network. The better user rating methods have higher average quality scores, demonstrating that the proposed quality network is effective.

8   Concluding Remarks

In this paper, we propose a method for automatic photo blending. To evaluate the aesthetic quality of blending photos, we collect a new dataset and design a quality network to learn from coarse user preferences with ranking loss and binary cross entropy. We tackle the photo blending problem by designing a deep reinforcement learning based agent to search for background alignment and photometric adjustment parameters that result in the highest score of the quality network. The proposed method can serve as a generic framework for automatic photo art generation.

References

1. Bychkovsky, V., Paris, S., Chan, E., Durand, F.: Learning photographic global tonal adjustment with a database of input/output image pairs. In: CVPR (2011)
2. Caicedo, J.C., Lazebnik, S.: Active object localization with deep reinforcement learning. In: ICCV (2015)
3. Chen, Q., Koltun, V.: Photographic image synthesis with cascaded refinement networks. In: ICCV (2017)
4. Chopra, S., Hadsell, R., LeCun, Y.: Learning a similarity metric discriminatively, with application to face verification. In: CVPR (2005)
5. Christiano, P., Leike, J., Brown, T.B., Martic, M., Legg, S., Amodei, D.: Deep reinforcement learning from human preferences. In: NIPS (2017)
6. Dumoulin, V., Shlens, J., Kudlur, M., Behboodi, A., Lemic, F., Wolisz, A., Molinaro, M., Hirche, C., Hayashi, M., Bagan, E., et al.: A learned representation for artistic style. In: ICLR (2017)
7. Finn, C., Levine, S., Abbeel, P.: Guided cost learning: Deep inverse optimal control via policy optimization. In: ICML (2016)
8. Gatys, L.A., Ecker, A.S., Bethge, M.: Image style transfer using convolutional neural networks. In: CVPR (2016)
9. Gharbi, M., Chen, J., Barron, J.T., Hasinoff, S.W., Durand, F.: Deep bilateral learning for real-time image enhancement (2017)
10. Girshick, R.: Fast r-cnn. In: ICCV (2015)
11. Guzman-Rivera, A., Batra, D., Kohli, P.: Multiple choice learning: Learning to produce multiple structured outputs. In: NIPS (2012)
12. Ho, J., Ermon, S.: Generative adversarial imitation learning. In: NIPS (2016)
13. Huang, X., Belongie, S.: Arbitrary style transfer in real-time with adaptive instance normalization. In: CVPR (2017)
14. Hwang, S.J., Kapoor, A., Kang, S.B.: Context-based automatic local image enhancement. In: ECCV. Springer (2012)
15. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. In: SIGGRAPH (2016)
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR (2017)
17. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: ECCV (2016)
18. Kang, L., Ye, P., Li, Y., Doermann, D.: Convolutional neural networks for no-reference image quality assessment. In: CVPR (2014)
19. Kennedy, J.: Particle swarm optimization. In: Encyclopedia of machine learning, pp. 760–766. Springer (2011)
20. Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P.: Optimization by simulated annealing. science **220**(4598), 671–680 (1983)
21. Kong, S., Shen, X., Lin, Z., Mech, R., Fowlkes, C.: Photo aesthetics ranking network with attributes and content adaptation. In: ECCV. Springer
22. Koyama, Y., Sato, I., Sakamoto, D., Igarashi, T.: Sequential line search for efficient visual design optimization by crowds. ACM TOG (Proc. SIGGRAPH) (2017)
23. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. In: ECCV (2016)
24. Lee, J.Y., Sunkavalli, K., Lin, Z., Shen, X., So Kweon, I.: Automatic content-aware color and tone stylization. In: CVPR (2016)

25. Li, C., Wand, M.: Combining markov random fields and convolutional neural networks for image synthesis. In: CVPR (2016)
26. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Diversified texture synthesis with feed-forward networks. In: CVPR (2017)
27. Li, Y., Fang, C., Yang, J., Wang, Z., Lu, X., Yang, M.H.: Universal style transfer via feature transforms. In: NIPS (2017)
28. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: NIPS (2017)
29. Liu, S., Pan, J., Yang, M.H.: Learning recursive filters for low-level vision via a hybrid neural network. In: ECCV (2016)
30. Lu, X., Lin, Z., Jin, H., Yang, J., Wang, J.Z.: Rapid: Rating pictorial aesthetics using deep learning. In: ACM MM (2014)
31. Lu, X., Lin, Z., Shen, X., Mech, R., Wang, J.Z.: Deep multi-patch aggregation network for image style, aesthetics, and quality estimation. In: ICCV (2015)
32. Luan, F., Paris, S., Shechtman, E., Bala, K.: Deep photo style transfer. In: CVPR (2017)
33. Mai, L., Jin, H., Liu, F.: Composition-preserving deep photo aesthetics assessment. In: CVPR (2016)
34. Marchesotti, L., Perronnin, F., Larlus, D., Csurka, G.: Assessing the aesthetic quality of photographs using generic image descriptors. In: ICCV (2011)
35. Mnih, V., Badia, A.P., Mirza, M., Graves, A., Lillicrap, T., Harley, T., Silver, D., Kavukcuoglu, K.: Asynchronous methods for deep reinforcement learning. In: ICML (2016)
36. Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., Graves, A., Riedmiller, M., Fidjeland, A.K., Ostrovski, G., et al.: Human-level control through deep reinforcement learning. Nature (2015)
37. Ng, A.Y., Russell, S.J., et al.: Algorithms for inverse reinforcement learning. In: ICML (2000)
38. Pathak, D., Krahenbuhl, P., Donahue, J., Darrell, T., Efros, A.A.: Context encoders: Feature learning by inpainting. In: CVPR (2016)
39. Ren, J., Shen, X., Lin, Z., Mech, R., Foran, D.J.: Personalized image aesthetics. In: ICCV (2017)
40. Shih, Y., Paris, S., Barnes, C., Freeman, W.T., Durand, F.: Style transfer for headshot portraits. In: SIGGRAPH (2014)
41. Silver, D., Huang, A., Maddison, C.J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al.: Mastering the game of go with deep neural networks and tree search. Nature (2016)
42. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: ICLR (2015)
43. Stadie, B.C., Abbeel, P., Sutskever, I.: Third-person imitation learning. ICLR (2017)
44. Tsai, Y.H., Shen, X., Lin, Z., Sunkavalli, K., Yang, M.H.: Sky is not the limit: Semantic-aware sky replacement. In: SIGGRAPH (2016)
45. Tsai, Y.H., Shen, X., Lin, Z., Sunkavalli, K., Lu, X., Yang, M.H.: Deep image harmonization. In: CVPR (2017)
46. Van Hasselt, H., Guez, A., Silver, D.: Deep reinforcement learning with double q-learning. In: AAAI (2016)
47. Wang, J.X., Kurth-Nelson, Z., Tirumala, D., Soyer, H., Leibo, J.Z., Munos, R., Blundell, C., Kumaran, D., Botvinick, M.: Learning to reinforcement learn. arXiv preprint arXiv:1611.05763 (2016)

48. Wang, Z., Schaul, T., Hessel, M., Van Hasselt, H., Lanctot, M., De Freitas, N.: Dueling network architectures for deep reinforcement learning. In: arXiv preprint arXiv:1511.06581 (2015)
49. Wei, Z., Zhang, J., Shen, X., Lin, Z., Mech, R., Hoai, M., Samaras, D.: Good view hunting: Learning photo composition from dense view pairs. In: CVPR (2018)
50. Yan, Z., Zhang, H., Wang, B., Paris, S., Yu, Y.: Automatic photo adjustment using deep neural networks. ACM TOG (Proc. SIGGRAPH) (2016)
51. Yeh, R., Chen, C., Lim, T.Y., Hasegawa-Johnson, M., Do, M.N.: Semantic image inpainting with perceptual and contextual losses. In: CVPR (2017)
52. Yoo, D., Park, S., Lee, J.Y., Paek, A.S., So Kweon, I.: Attentionnet: Aggregating weak directions for accurate object detection. In: ICCV (2015)
53. Yoo, S.Y.J.C.Y., Yun, K., Choi, J.Y.: Action-decision networks for visual tracking with deep reinforcement learning. In: CVPR (2017)
54. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. In: ECCV (2016)
55. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep networks as a perceptual metric. In: CVPR (2018)
56. Zhang, R., Zhu, J.Y., Isola, P., Geng, X., Lin, A.S., Yu, T., Efros, A.A.: Real-time user-guided image colorization with learned deep priors. In: SIGGRAPH (2017)
57. Zhu, J.Y., Krahenbuhl, P., Shechtman, E., Efros, A.A.: Learning a discriminative model for the perception of realism in composite images. In: ICCV (2015)
58. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV (2017)
59. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: NIPS (2017)