

EMEF: Ensemble Multi-Exposure Image Fusion

Renshuai Liu, Chengyang Li, Haitao Cao, Yinglin Zheng, Ming Zeng, Xuan Cheng*

School of Informatics, Xiamen University, Xiamen 361005, China
 {medalwill, chengyanglee, chtao, zhengyinglin}@stu.xmu.edu.cn,
 {zengming, chengxuan}@xmu.edu.cn

Abstract

Although remarkable progress has been made in recent years, current multi-exposure image fusion (MEF) research is still bounded by the lack of real ground truth, objective evaluation function, and robust fusion strategy. In this paper, we study the MEF problem from a new perspective. We don't utilize any synthesized ground truth, design any loss function, or develop any fusion strategy. Our proposed method EMEF takes advantage of the wisdom of multiple imperfect MEF contributors including both conventional and deep learning-based methods. Specifically, EMEF consists of two main stages: pre-train an imitator network and tune the imitator in the runtime. In the first stage, we make a unified network imitate different MEF targets in a style modulation way. In the second stage, we tune the imitator network by optimizing the style code, in order to find an optimal fusion result for each input pair. In the experiment, we construct EMEF from four state-of-the-art MEF methods and then make comparisons with the individuals and several other competitive methods on the latest released MEF benchmark dataset. The promising experimental results demonstrate that our ensemble framework can "get the best of all worlds". The code is available at <https://github.com/medalwill/EMEF>.

1 Introduction

Real-world scenes usually exhibit a high dynamic range (HDR) that may be in excess of 100,000: 1 between the brightest and darkest regions. The pictures captured by digital image sensors, however, usually have a low dynamic range (LDR), suffering from over-exposure and under-exposure in some situations. An effective yet economical solution is MEF, which fuses several LDR images in different exposures into a single HDR image. Nowadays, MEF has already been widely used in smartphones like Xiaomi, vivo and OPPO.

Many MEF methods have been proposed over the last decade. The traditional MEF methods use specific hand-crafted fusion strategies, while the deep learning-based MEF methods directly feed multi-exposure images into a network to produce a fused image in a supervised or unsupervised way. Although deep learning-based methods have gradually

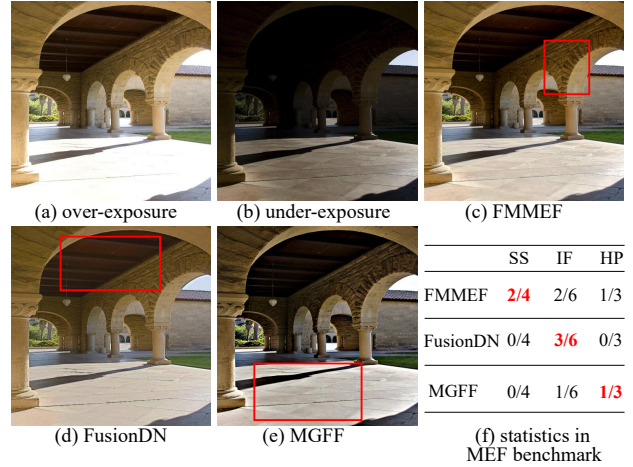


Figure 1: The over-exposure (a) and under-exposure (b) images are the input. The fusion results of FMMEF (c), FusionDN (d), and MGFF (e) respectively exhibit more meaningful structures in the stone arch, roof, and ground regions. According to the evaluation (f) in MEFB (Zhang 2021), FMMEF, FusionDN, and MGFF have advantages respectively in the metrics of structural similarity (SS), image feature (IF) and human perception (HP). "2/4" means that the method gets 2 best values among the 4 metrics.

become mainstream in the MEF field, traditional methods still show very competitive performance in a recently published MEF benchmark (MEFB) (Zhang 2021). Meanwhile, it's really hard to find a perfect MEF method at present, as no single method can always perform well in all situations. This is mainly due to three factors. 1) The existing MEF ground truth data is mostly artificially made by selecting visually appealing results from a set of MEF methods. The lack of real ground truth hinders the ability of learning-based methods. 2) As HDR is a very subjective visual effect of human beings, there is no uniform objective metric that can well evaluate the fusion quality. Hence, the loss functions used in existing MEF methods are usually biased. 3) Most traditional methods make assumptions about the scenes, which are valid in some situations but invalid in others. It's really hard to design a one-size-fits-all image fusion strategy.

*corresponding author

As a consequence, the existing MEF methods have their own strengths and weaknesses. Based on the comprehensive quantitative evaluation of MEFB (Zhang 2021), FMMEF (Li et al. 2020), FusionDN (Xu et al. 2020b) and MGFF (Bavirisetti et al. 2019) are the top three methods. FMMEF performs well in structural similarity-based metrics, FusionDN exhibits good performance in image feature-based metrics, and MGFF gets high scores in human perception-inspired metrics. This phenomenon clearly dedicates that current state-of-the-arts have unique advantages when examined from different aspects. We show an example in Fig. 1.

In this paper, we study the MEF problem from a fire-new perspective. We don’t utilize any synthesized ground truth, design any loss function, or develop any fusion strategy, like other traditional or deep learning-based MEF methods. Our proposed method takes advantage of the wisdom of multiple imperfect MEF methods, by combining each method’s solution for the problem to give a higher quality solution than any individual. We refer to our method as *ensemble-based MEF* (EMEF), as it shares a similar motivation with other ensemble methods (Bai et al. 2013; Wang and Yeung 2014) that combine multiple models.

The main contribution of this paper is the ensemble framework for the MEF problem. To realize the framework, we also propose several new network designs: 1) the imitator network which imitates different MEF methods’ fusion effect in the unified GAN framework; 2) the optimization algorithm which searches the optimal code in the style space of the imitator to make MEF inference; 3) the random soft label to represent the style code, which removes artifacts while improves generalization.

2 Related Work

2.1 Traditional MEF Methods

Traditional MEF methods generally consist of spatial domain-based methods and transform domain-based methods.

Spatial domain-based methods can be further divided into three categories, i.e., pixel-based, patch-based, and optimization-based methods. Pixel-based methods work on the pixel level, calculating the weighted sum of source images to derive a fused image. DSIFT-EF (Liu and Wang 2015) estimates the weight maps of source images according to their local contrast, exposure quality, and spatial consistency, then refines the weight maps by a recursive filter. MEFAW (Lee, Park, and Cho 2018) defines two adaptive weights based on the relative intensity and global gradient of each pixel. The final weight maps are worked out with a normalized multiplication operation. Different from the pixel-wised methods, patch-wised methods work on the patch level. The method proposed by (Ma and Wang 2015) decomposes each patch into signal strength, signal structure, and mean intensity, then reconstructs patches with the above components, and finally blends them to generate a fused image. Based on the above method, SPD-MEF (Ma et al. 2017) makes use of the direction of the signal structure component to achieve ghost removal. The representa-

tive of optimization-based methods is MEFOpt (Ma et al. 2018). This method introduces an evaluation metric named MEF-SSIM_c which has improved performance compared to the original MEF-SSIM metric. Then the gradient descent is used to search the space of all images for a fusion result with optimal performance on MEF-SSIM_c.

Transform domain-based methods firstly transform source images to a specific domain to get their implicit representations, then fuse these representations, and finally convert the fusion results back to the spatial domain by an inverse transform. The method proposed by (Burt and Kolczynski 1993) is one of the first transform domain-based MEF methods which uses a gradient pyramid transform to get pyramid representations. The method proposed by (Mertens, Kautz, and Reeth 2007) estimates the weight maps of source images considering their contrast, saturation, and exposure followed by a Gaussian filter smoothing. It also adopts a Laplacian pyramid transform to get laplacian coefficients which are then weighted according to the weight maps. Based on the above method, (Li, Manjunath, and Mitra 1995) fuse source images in the wavelet domain after a wavelet transform.

Although traditional methods have made great progress, they still have some drawbacks. e.g., it’s not an easy task to design an effective fusion algorithm and there is no one-size-fits-all fusion strategy.

2.2 Deep Learning-Based MEF Methods

Deep learning-based methods usually train networks in a supervised or unsupervised manner. As real ground truth data is hard to obtain, researchers attempt to synthesize ground truth by various means. EFCNN (Wang et al. 2018) is one of the earliest supervised methods which makes ground truth by adjusting the pixel intensity of source images. SICE (Cai, Gu, and Zhang 2018) establishes a paired dataset by generating fusion results of existing methods and manually selecting the visually best one as ground truth. Based on the synthesized ground truth, MEF-GAN (Xu, Ma, and Zhang 2020) makes progress by introducing GAN and self-attention mechanism into the field of MEF. CF-Net (Deng et al. 2021) suggests undertaking the super-resolution task and MEF task with a unified network so that collaboration and interaction between them can be achieved.

Another kind of method trains their networks in other tasks to learn image representations of source images, and then fuse these representations to reconstruct the final result. IFCNN (Zhang et al. 2020b) trains its model on a multi-focus fusion dataset. Similarly, transMEF (Qu et al. 2022) applies three self-supervised image reconstruction tasks to capture the representations of source images.

Unsupervised methods take a different way to work without ground truth. They fuse the source images under the guidance of a specific image assessment metric. DeepFuse (Prabhakar, Srikanth, and Babu 2017) is not only the first unsupervised method but also the first deep-learning method, which works on YCrCb color space and applies MEF-SSIM to train its CNN. DIF-Net (Jung et al. 2020) is designed for several image fusion tasks which focus on contrast preservation by employing a metric named structure tensor.

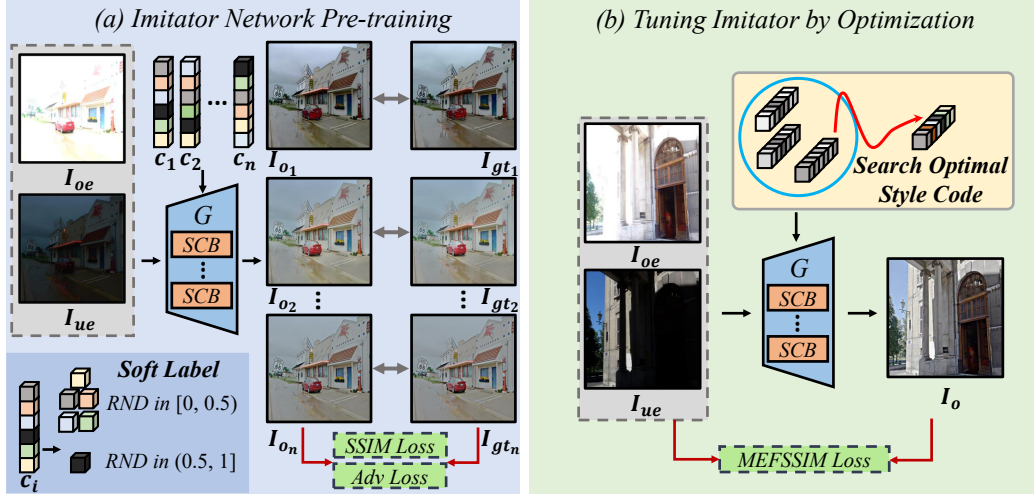


Figure 2: Overview. The proposed EMEF consists of two main stages: (a) pre-train an imitator network, and (b) tune the imitator in the runtime.

U2Fusion (Xu et al. 2020a) measures the amount and quality of the information in source images by computing information preservation degree and adaptively fuses source images with respect to it. PMGI (Zhang et al. 2020a) considers the fusion task as a proportional maintenance problem of gradient and intensity, which utilizes a two-branch network and divides the loss function into intensity and gradient parts.

3 Method

3.1 Overview

As shown in Fig. 2, the proposed EMEF consists of two main stages: pre-train an imitator network (Sect. 3.2), and tune the imitator in the runtime (Sect. 3.3). In the first stage, we utilize a unified network to imitate multiple MEF methods. Several traditional and deep learning-based MEF methods are selected as the imitation targets. We view each MEF target method as a “style” and then train a style-modulated GAN in a supervised way. Such a network can produce a very similar fusion result with each target method in the ensemble, under the control of a style code. The style code determines which target methods the network would imitate in the online inference, and is represented by the random soft label (Sect. 3.4) to improve the generalization ability. In the second stage, we tune the pre-trained imitator network by searching the optimal style code, in order to make inferences for each input pair. An image quality assessment-based loss function is optimized in the gradient descent way. Finally, EMEF is able to produce the best fusion result from the combined space of the MEF target methods.

3.2 Imitator Network Pre-training

Before constructing the imitator network, we collect the training data for it. For a pair of over-exposed and under-exposed images denoted by I_{oe} , I_{ue} , we use all the MEF target methods $\mathcal{M}_i, i = 1, 2, \dots, n$ in the ensemble to produce their fusion images I_{gt_i} . Besides, each \mathcal{M}_i is represented by

a certain style code $c_i \in \mathbb{R}^n$. A sample in the training data can be formulated as:

$$[I_{oe}, I_{ue}, \{I_{gt_i}, c_i\}_{i=1,2,\dots,n}]. \quad (1)$$

We believe that the particular fusion strategy and loss functions used in \mathcal{M}_i have already been embedded in our constructed training data, and can be learned by the deep models.

The core of the imitator network is a style-modulated generator denoted by \mathcal{G} . As shown in Fig. 2, the generator takes the image pair I_{oe} , I_{ue} , and the style code c_i as input, and outputs the fusion result I_{o_i} . Such generation process can be formulated as:

$$I_{o_i} = \mathcal{G}(I_{oe}, I_{ue}, c_i, \theta), \quad (2)$$

where θ is the parameters of \mathcal{G} . We require I_{o_i} to match its corresponding I_{gt_i} as much as possible and thus train \mathcal{G} in a supervised manner.

The network architecture of \mathcal{G} is shown in Fig. 3, which adopts a standard UNet as the backbone and incorporates the Style Control Block (SCB). The UNet extracts multi-scale features from the input images in the encoder and adds them back to each layer in the decoder, which helps to preserve more information from the input. The SCB injects the style code c_i to each layer in the decoder of UNet except the last one, which is the key to our style control. The style code is not directly used but mapped into a latent space by a Multi-layer Perceptron (MLP) before being fed to SCB.

Style Control Block. We leverage the merit of StyleGAN2 (Karras et al. 2020) to construct SCB. SCB consists of a convolution layer and two operations (modulation and demodulation) to its weights. For an input latent code l , SCB firstly transforms the l_i of the i th layer into the micro style s_i with an affine transformation, so that the latent code l can match the scale of different layers. Then the weights are scaled with the s_i to fuse s_i into the activation, which helps to decouple styles of different target methods. The weight

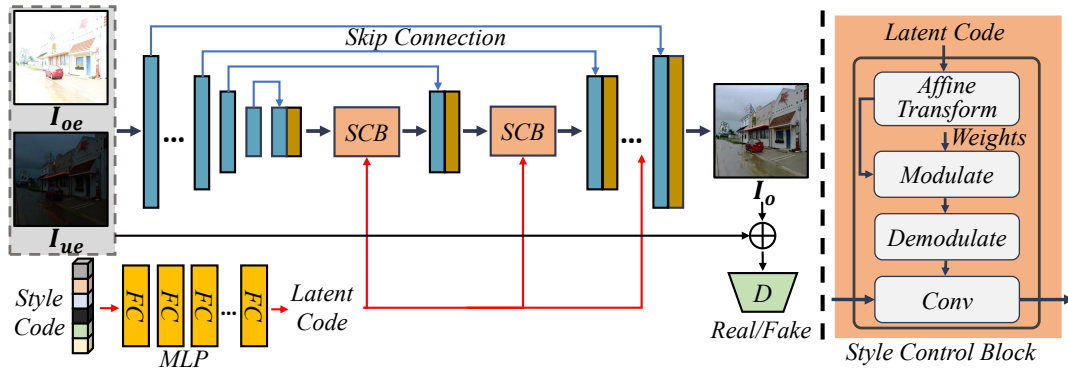


Figure 3: Generator. The network architecture consists of the UNet, several Style Control Blocks (SCB), and the MLP-based mapping block.

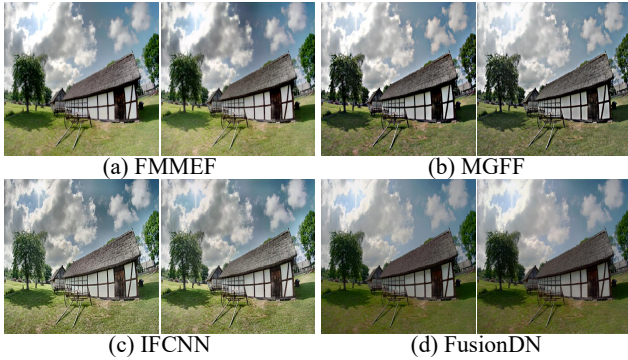


Figure 4: An example of our MEF imitation. In each pair, the left one is the fusion result I_{gt_i} from the imitation target method while the right one is the imitative result I_{o_i} from our imitator network.

modulation operation can be formulated as:

$$w'_{ijkl} = s_j \cdot w_{ijkl}, \quad (3)$$

where w denotes the original weights, w' denotes the modulated weights, i denotes the i -th output channel of the weight, j denotes the j -th input channel of the weight, and (k, l) denotes the coordinate of the convolution kernel. Subsequently, weight demodulation is conducted, which shrinks the weights to keep the statistics of activations unchanged. It's formulated as:

$$w''_{ijkl} = w'_{ijkl} / \sqrt{\sum_{j,k,l} w'_{ijkl}{}^2 + \epsilon} \quad (4)$$

where ϵ is a small positive constant to promote robustness.

Loss. The imitator network is optimized by minimizing SSIM loss and adversarial loss. SSIM loss measures the structural similarity between I_{o_i} and its corresponding ground truth I_{gt_i} , which can be formulated as:

$$L_{SSIM} = 1 - SSIM(I_{o_i}, I_{gt_i}). \quad (5)$$

$SSIM(\cdot, \cdot)$ denotes the standard SSIM metric. To promote the realism of I_{o_i} , we also employ adversarial loss. It's for-

mulated as:

$$L_{adv} = \mathbb{E}(1 - \log \mathcal{D}(I_{o_i}, I_{oe}, I_{ue}, \gamma)) + \mathbb{E}(\log \mathcal{D}(I_{gt_i}, I_{oe}, I_{ue}, \gamma)), \quad (6)$$

where \mathcal{D} is the discriminator and γ is its parameters.

The final loss is the weighted sum of the aforementioned losses:

$$L_{pre} = L_{SSIM} + \lambda L_{adv} \quad (7)$$

where λ is a trade-off between the two losses. Fig. 4 shows our imitation results. There is little visual difference between the output I_{o_i} of our imitator network and the fusion result of the target method I_{gt_i} .

3.3 Imitator Network Tuning

Algorithm 1: Search for the optimal style code $c^* \in \mathbb{R}^n$

Input: A pair of over-exposed and under-exposed images I_{oe}, I_{ue} . The pre-trained imitator network \mathcal{G} .

Initialize: Initialize the style code c^* with an all-one vector. Concatenate I_{oe}, I_{ue} into I_{oue} in the channel dimension.

- 1: **repeat**
- 2: $L \leftarrow 1 - MEFSSIM(I_{oue}, \mathcal{G}(I_{oe}, I_{ue}, c^*))$
- 3: $c^* \leftarrow c^* - \alpha \nabla_{c^*} L$
- 4: **until** converged

Output: c^*

In this stage, we tune the style code in the pre-trained imitator network to make inferences for an input pair. As we mentioned before, there is no perfect MEF method. It's better to utilize different suitable MEF methods for different types of source images. To realize the goal, we search for an optimal style code for the input pair. The pseudo-code of the searching procedure is presented in Algorithm 1. Starting from an all-one initialization, we use the gradient descent algorithm to search for an optimal style code c^* that minimizes the MEF-SSIM (Ma, Zeng, and Wang 2015) loss function. The MEF-SSIM measures how much vital information from input images can be preserved in the fused image, and is a frequently used MEF image quality assessment model.

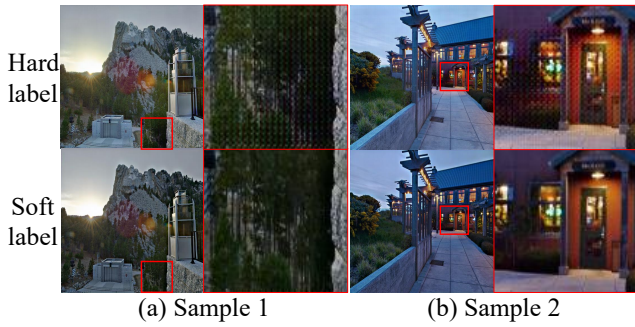


Figure 5: Hard label introduces unacceptable image artifacts, such as stripes and grids, while random soft label can remove them by mitigating the domain gap.

3.4 Random Soft Label

Intuitively, we use the one-hot label (e.g. $\{0, 1, 0, 0\}$) as the style code in the imitator network pre-training. However, the optimized style codes obtained in the imitator tuning stage are usually floats (e.g. $\{0.22, 0.15, 0.85, 0.36\}$) rather than integers. The significant domain gap between the style codes in pre-training and tuning introduces severe artifacts. We show an example in Fig. 5. To overcome this issue, we adopt a random soft label trick that can mitigate the domain gap and promote robustness. We replace the 1 value in the one-hot label with a random number in the range of $(0.5, 1.0]$, and replace the 0 value with a random number in the range of $[0.0, 0.5)$. For example, $\{0, 1, 0, 0\}$ is replaced by $\{0.08, 0.73, 0.36, 0.21\}$. The experimental results demonstrate that the random soft label eliminates artifacts, mitigates the domain gap and greatly improves the generalization ability of the network.

3.5 Discussions

Relationship with supervised MEF. The supervised MEF methods (Wang et al. 2018; Cai, Gu, and Zhang 2018; Xu, Ma, and Zhang 2020) directly optimize the reconstruction loss between the fusion result and the ground truth. However, the ground truth is usually artificially made. Our proposed EMEF uses the supervised way only for imitating different MEF targets in a unified framework, but not for generating the fusion result.

Relationship with unsupervised MEF. The unsupervised MEF methods (Prabhakar, Srikanth, and Babu 2017; Xu et al. 2020a) usually optimize the loss that measures the retention degrees of image features from the input. We also use such unsupervised loss in the MEF inference. The unsupervised method searches the entire image space, while our proposed EMEF takes the pre-trained imitator network as the prior, thus constructing a smaller space (the combined space of the MEF target methods). Hence, EMEF searches for a low-dimension style code rather than a high-dimension image, which provides the one-shot MEF and increases robustness.

Relationship with ensemble GAN. There are ensemble methods (Arora et al. 2017; Hoang et al. 2018; Han, Chen, and Liu 2021) that train GAN with multiple gener-

ators rather than a single one, thus delivering a more stable image generation. Our method uses a unified generator to model different data distributions of the fusion results in different MEF methods. Another difference is that the ensemble GANs usually produce the final result by averaging or randomly selecting the generators’ output, while our method finds an optimal result by optimization in the generator’s style space.

4 Experiments

4.1 Implementation Details

In our experiments, λ is set to 0.002. The network architecture of the generator follows the image-to-image translation network (Isola et al. 2017). The image size of both input and output are 512×512 . In the imitator network pre-training, the batch size is set to 1 and the network is trained with an Adam optimizer for 100 epochs. In the first 50 epochs, the learning rate is set to $2e - 4$, and then decays linearly for the rest. In the imitator tuning, we adopt an adaptive search strategy with a 20-step linear learning rate decay. We choose the top-four MEF methods in MEFB to construct the ensemble, including FMMEF, FusionDN, MGFF, and IFCNN, and implement EMEF with Pytorch. All experiments are conducted with two GeForce RTX 3090 GPUs. It takes about 1.8 minutes to generate a 512×512 fusion image.

4.2 Experimental Settings

Datasets. We train EMEF with the SICE (Cai, Gu, and Zhang 2018) dataset and evaluate it in MEFB (Zhang 2021). SICE contains 589 image sequences of different exposures, and each sequence has a selected fused image attached as ground truth. We focus on the extreme static MEF problem so that only the brightest image and the darkest image within 356 static scene sequences are selected as the training data. MEFB contains 100 over-exposure and under-exposure image pairs captured under various conditions, which can provide fair and comprehensive comparisons.

Evaluation Metrics. We apply 12 metrics from four perspectives to evaluate the proposed EMEF. The metrics include information theory-based metrics, CE (Bulanon, Burks, and Alchanatis 2009), EN (Roberts, van Aardt, and Ahmed 2008), PSNR (Jagalingam and Hegde 2015), TE (Cvejic, Canagarajah, and Bull 2006); image feature-based metrics, AG (Cui et al. 2015), EI (Rajalingam and Priya 2018), $Q^{AB/F}$ (Xydeas and Petrovic 2000), Q_P (Zhao, Laganier, and Liu 2007), SF (Eskicioglu and Fisher 1995); structural similarity-based metrics, Q_W (Piella and Heijmans 2003), MEF – SSIM (Ma, Zeng, and Wang 2015); human perception inspired metrics, Q_{CV} (Chen and Varshney 2007).

Competitors. Besides the four MEF methods included in the ensemble, we choose other 6 competitive MEF methods as competitors, which include three traditional methods, DSIFT-EF (Liu and Wang 2015), SPD-MEF (Ma et al. 2017), MEFopt (Ma et al. 2018); and three deep learning-based methods, MEF-GAN (Xu, Ma, and Zhang 2020), U2Fusion (Xu et al. 2020a), transMEF (Qu et al. 2022).

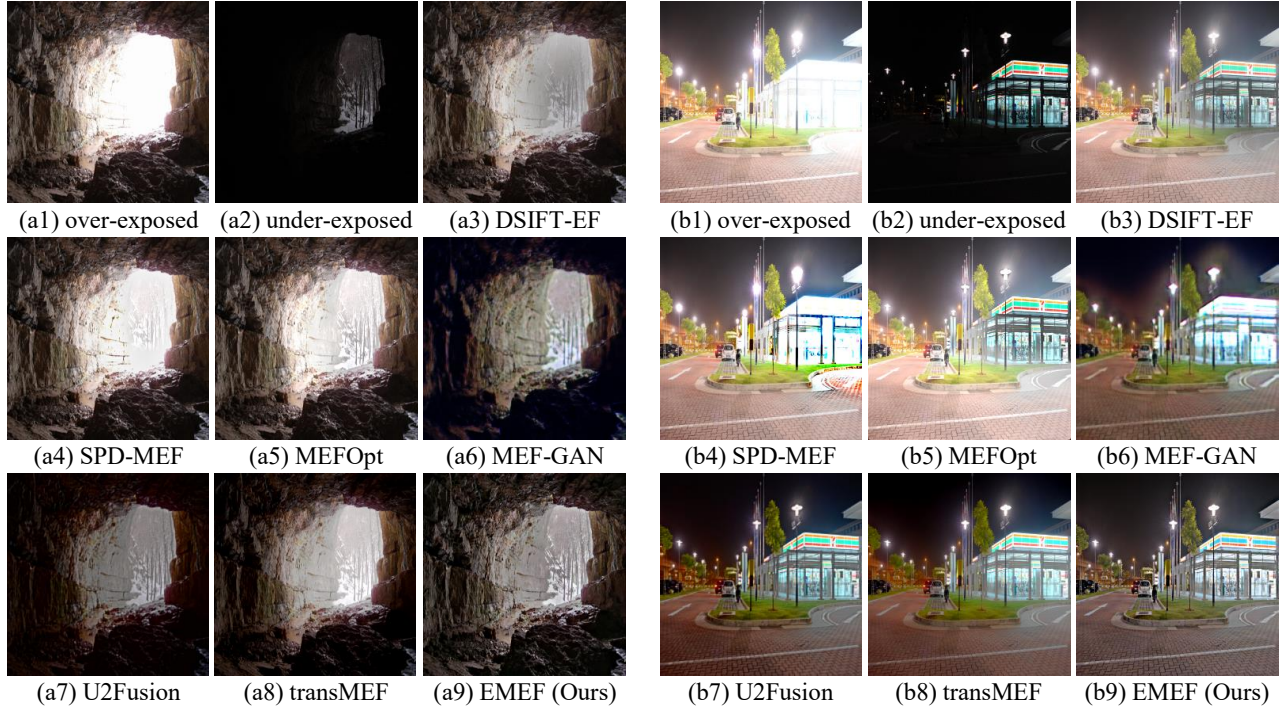


Figure 6: Qualitative comparison of EMEF with 6 competitive MEF methods on 2 typical multi-exposure image pairs in the MEFB dataset.

Methods	CE ↓	EN ↑	PSNR ↑	TE ↑	AG ↑	EI ↑	$Q^{AB/F} ↑$	$Q_P ↑$	SF ↑	$Q_W ↑$	MEF-SSIM ↑	$Q_{CV} ↓$	Overall Rank ↓
DSIFT-EF	1.3026(1)	7.261(1)	53.336(6)	9866.552(3)	5.836(7)	59.269(7)	0.740(1)	0.769(1)	18.836(6)	0.865(3)	0.855(3)	519.914(6)	45
SPD-MEF	2.6909(6)	7.112(4)	53.594(3)	8801.729(5)	6.890(3)	69.691(3)	0.683(4)	0.737(2)	23.086(1)	0.822(6)	0.833(5)	383.757(4)	46
MEFOpt	2.3232(5)	7.195(3)	53.231(7)	11489.361(2)	6.924(2)	69.939(2)	0.731(2)	0.698(5)	22.188(3)	0.899(1)	0.866(2)	578.273(7)	41
MEF-GAN	1.8621(3)	6.933(5)	53.520(5)	7130.901(7)	5.874(6)	62.075(5)	0.442(7)	0.388(7)	17.308(7)	0.611(7)	0.761(7)	394.722(5)	71
U2Fusion	1.9372(4)	6.609(7)	53.621(2)	20056.882(1)	5.986(4)	62.349(4)	0.596(6)	0.654(6)	19.021(5)	0.826(5)	0.849(4)	279.322(2)	50
transMEF	2.7162(7)	6.768(6)	53.581(4)	8393.075(6)	5.891(5)	59.780(6)	0.682(5)	0.722(4)	19.648(4)	0.836(4)	0.832(6)	247.126(1)	58
EMEF	1.7607(2)	7.219(2)	53.624(1)	9134.513(4)	6.969(1)	70.177(1)	0.693(3)	0.725(3)	22.755(2)	0.885(2)	0.875(1)	312.892(3)	25

Table 1: Quantitative results of EMEF and several MEF competitors (DSIFT-EF, SPD-MEF, MEFOpt, MEF-GAN, U2Fusion, transMEF) over MEFB in 256×256 resolution. All metrics except CE and Q_{cv} follow “higher is better”. The number listed within the bracket after each score denotes the rank in the metric. The overall rank is the sum of ranks on all metrics.

Methods	CE ↓	EN ↑	PSNR ↑	TE ↑	AG ↑	EI ↑	$Q^{AB/F} ↑$	$Q_P ↑$	SF ↑	$Q_W ↑$	MEF-SSIM ↑	$Q_{CV} ↓$	Overall Rank ↓
FMMEF	2.7538(6)	7.146(7)	56.557(5)	16371.098(8)	5.052(9)	51.808(9)	0.765(1)	0.760(1)	16.951(9)	0.914(1)	0.893(3)	417.983(7)	66
MGFF	2.8829(9)	7.088(8)	56.603(2)	7914.649(9)	6.096(2)	62.546(2)	0.692(3)	0.740(2)	20.502(2)	0.860(4)	0.884(6)	346.728(1)	50
IFCNN	2.8488(8)	7.303(1)	56.422(7)	37365.314(3)	8.190(1)	80.417(1)	0.562(9)	0.618(6)	26.253(1)	0.791(8)	0.842(9)	450.971(8)	62
FusionDN	2.7072(5)	7.242(4)	56.397(8)	56725.167(2)	5.375(7)	55.559(7)	0.589(8)	0.588(7)	16.979(8)	0.789(9)	0.868(8)	371.430(4)	77
Pick I_{gt}^*	2.7665(7)	7.181(6)	56.535(6)	18988.347(7)	5.368(8)	55.086(8)	0.720(2)	0.721(3)	17.722(7)	0.883(2)	0.887(5)	372.248(5)	66
Pick I_o^*	1.7470(1)	7.271(3)	56.563(4)	19542.253(6)	5.590(4)	57.215(4)	0.640(5)	0.654(5)	18.661(4)	0.844(7)	0.889(4)	400.537(6)	53
opt. latent code	2.1209(4)	6.774(9)	56.384(9)	81545.751(1)	5.565(5)	56.158(5)	0.610(7)	0.477(9)	17.934(6)	0.850(6)	0.955(1)	714.371(9)	71
w/o. soft label	1.9037(3)	7.272(2)	56.584(3)	28799.171(4)	6.054(3)	60.470(3)	0.613(6)	0.585(8)	19.542(3)	0.854(5)	0.873(7)	370.834(3)	50
EMEF	1.9035(2)	7.239(5)	56.618(1)	22898.593(5)	5.504(6)	56.130(6)	0.667(4)	0.665(4)	18.319(5)	0.876(3)	0.898(2)	365.065(2)	45

Table 2: Quantitative results of EMEF, the methods in the ensemble (FMMEF, MGFF, IFCNN, FusionDN), and the methods in ablation study (pick I_{gt}^* , pick I_o^* , optimize latent code, w/o. soft label) over MEFB in 512×512 resolution.

4.3 Comparisons with the MEF Competitors

The qualitative comparison of EMEF with the 6 competitors is shown in Fig. 6. The main goal of MEF is to make the dark region brighter while making the bright region darker so that more details can be maintained. In sample (a), the region inside the cave is dark and the region outside is bright. SPD-MEF, MEFopt failed to darken the bright region while MEF-GAN, U2Fusion was unable to brighten the dark region. DSIFT-EF and transMEF manage to behave well in both regions but exhibit lower contrast and fewer details outside the cave compared with our EMEF. In sample (b), the goal is to provide appropriate exposure for the streetlights, the store, and the floor. DSIFT-EF, SPD-MEF, MEFopt, and MEF-GAN failed to achieve this goal, as terrible halo artifacts surround the streetlights and the store. Our method produces more appealing luminance in the left part of the fused image than U2Fusion. When compared with transMEF, our method generates far finer details in the store and the floor regions.

The quantitative comparison is presented in Table 1. The existing MEF methods are capable of achieving good performance on their preference metrics. Due to the extreme pursuit of these metrics, they usually show poor scores on the remaining metrics. The overall rank is the sum of ranks on all metrics which can reveal overall performance. We integrate 4 distinctive methods in EMEF so that they compensate each other. Thus our method has a relatively all-round ability on all metrics, ultimately presenting balanced and optimal overall performance.

4.4 Comparisons with the MEF Methods in the Ensemble

The qualitative comparison of EMEF with the 4 methods included in the ensemble is shown in Fig. 7. In sample (a), MGFF and FusionDN fail to brighten the dark regions in the sea, while FMMEF and IFCNN don't recover more details in the sky compared with ours. In sample (b), MGFF fails to brighten the dark region in the grass, FMMEF and IFCNN exhibit a little over-exposure in the grass, and FusionDN generates an image of low contrast and poor luminance. In both samples, our method can reconstruct the scene with moderate lighting conditions. The quantitative comparison is presented in Table 2. Our method surpasses the methods included in the ensemble in the overall performance due to its integrated capability.

4.5 Ablation Study

In the ablation study, we evaluate other four methods: 1) selecting the MEF result I_{gt}^* directly from the outputs of the four methods in the ensemble with the highest MEF-SSIM score, 2) selecting the MEF result I_o^* directly from the four imitation results of imitator network with the highest MEF-SSIM score, 3) optimizing the latent code instead of the style code, 4) replacing the soft label with the hard label. The quantitative results are presented in Table 2. It can be observed that the method of picking I_{gt}^* prefers to pick FMMEF since it ranks first in the MEFB. The method of picking I_o^* performs better than picking I_{gt}^* , which indicates

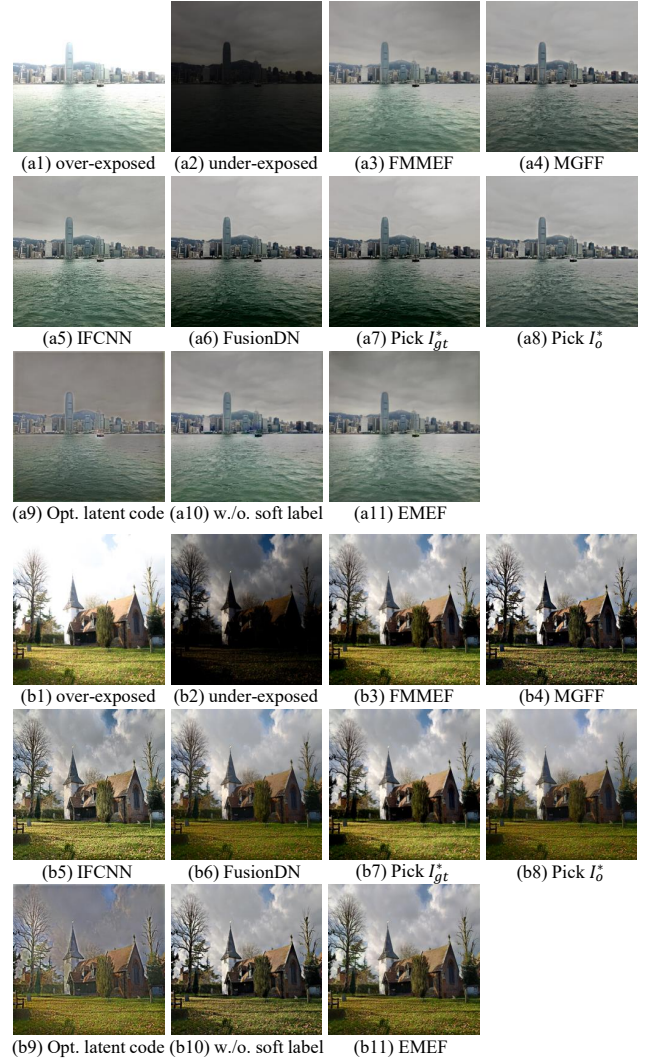


Figure 7: Qualitative comparison of EMEF with the four MEF methods in the ensemble and the four methods in the ablation study.

that the imitator network improves the individuals' ability while combining them. The method of optimizing the latent code behaves badly, as it introduces severe color distortion in the image. The method without soft label introduces artifacts that degrades the performance. Finally, our method of optimizing the style code performs the best, which demonstrates the effectiveness of our optimization and random soft label. The qualitative evaluation of the compared methods is shown in Fig. 7.

5 Conclusion

In this paper, we propose an ensemble-based MEF method, named EMEF. Extensive comparisons in the benchmark have provided evidence for the feasibility and effectiveness of EMEF. The ensemble framework also has the potential to be used in other image generation tasks.

Acknowledgments

This work was partially supported by NSFC (No. 61802322) and Natural Science Foundation of Xiamen, China (No. 3502Z20227012).

References

- Arora, S.; Ge, R.; Liang, Y.; Ma, T.; and Zhang, Y. 2017. Generalization and Equilibrium in Generative Adversarial Nets (GANs). In *Proc. of ICML*, volume 70, 224–232.
- Bai, Q.; Wu, Z.; Sclaroff, S.; Betke, M.; and Monnier, C. 2013. Randomized Ensemble Tracking. In *Proc. of ICCV*, 2040–2047.
- Bavirisetti, D. P.; Xiao, G.; Zhao, J.; Dhuli, R.; and Liu, G. 2019. Multi-scale Guided Image and Video Fusion: A Fast and Efficient Approach. *Circuits, Systems, and Signal Processing*, 38(12): 5576–5605.
- Bulanon, D. M.; Burks, T. F.; and Alchanatis, V. 2009. Image fusion of visible and thermal images for fruit detection. *Biosystems Engineering*, 103(1): 12–22.
- Burt, P.; and Kolczynski, R. 1993. Enhanced image capture through fusion. In *Proc. of ICCV*, 173–182.
- Cai, J.; Gu, S.; and Zhang, L. 2018. Learning a Deep Single Image Contrast Enhancer from Multi-Exposure Images. *IEEE Transactions on Image Processing*, 27(4): 2049–2062.
- Chen, H.; and Varshney, P. K. 2007. A human perception inspired quality metric for image fusion based on regional information. *Information Fusion*, 8(2): 193–207.
- Cui, G.; Feng, H.; Xu, Z.; Li, Q.; and Chen, Y. 2015. Detail preserved fusion of visible and infrared images using regional saliency extraction and multi-scale image decomposition. *Optics Communications*, 341: 199–209.
- Cvejic, N.; Canagarajah, C. N.; and Bull, D. R. 2006. Image fusion metric based on mutual information and Tsallis entropy. *Electronics Letters*, 42(11): 626–627.
- Deng, X.; Zhang, Y.; Xu, M.; Gu, S.; and Duan, Y. 2021. Deep coupled feedback network for joint exposure fusion and image super-resolution. *IEEE Transactions on Image Processing*, 30: 3098–3112.
- Eskicioglu, A. M.; and Fisher, P. S. 1995. Image quality measures and their performance. *IEEE Transactions on Communications*, 43(12): 2959–2965.
- Han, X.; Chen, X.; and Liu, L. 2021. GAN Ensemble for Anomaly Detection. In *Proc. of AAAI*, 4090–4097.
- Hoang, Q.; Nguyen, T. D.; Le, T.; and Phung, D. Q. 2018. MGAN: Training Generative Adversarial Nets with Multiple Generators. In *Proc. of ICLR*.
- Isola, P.; Zhu, J.; Zhou, T.; and Efros, A. A. 2017. Image-to-Image Translation with Conditional Adversarial Networks. In *Proc. of CVPR*, 5967–5976.
- Jagalingam, P.; and Hegde, A. V. 2015. A Review of Quality Metrics for Fused Image. *Aquatic Procedia*, 4: 133–142.
- Jung, H.; Kim, Y.; Jang, H.; Ha, N.; and Sohn, K. 2020. Unsupervised deep image fusion with structure tensor representations. *IEEE Transactions on Image Processing*, 29: 3845–3858.
- Karras, T.; Laine, S.; Aittala, M.; Hellsten, J.; Lehtinen, J.; and Aila, T. 2020. Analyzing and improving the image quality of stylegan. In *Proc. of CVPR*, 8107–8116.
- Lee, S.-h.; Park, J. S.; and Cho, N. I. 2018. A Multi-Exposure Image Fusion Based on the Adaptive Weights Reflecting the Relative Pixel Intensity and Global Gradient. In *Proc. of ICIP*, 1737–1741.
- Li, H.; Ma, K.; Yong, H.; and Zhang, L. 2020. Fast Multi-Scale Structural Patch Decomposition for Multi-Exposure Image Fusion. *IEEE Transactions on Image Processing*, 29: 5805–5816.
- Li, H.; Manjunath, B.; and Mitra, S. K. 1995. Multisensor image fusion using the wavelet transform. *Graphical Models and Image Processing*, 57(3): 235–245.
- Liu, Y.; and Wang, Z. 2015. Dense SIFT for ghost-free multi-exposure fusion. *Journal of Visual Communication and Image Representation*, 31: 208–224.
- Ma, K.; Duanmu, Z.; Yeganeh, H.; and Wang, Z. 2018. Multi-Exposure Image Fusion by Optimizing A Structural Similarity Index. *IEEE Transactions on Computational Imaging*, 4(1): 60–72.
- Ma, K.; Li, H.; Yong, H.; Wang, Z.; Meng, D.; and Zhang, L. 2017. Robust Multi-Exposure Image Fusion: A Structural Patch Decomposition Approach. *IEEE Transactions on Image Processing*, 26(5): 2519–2532.
- Ma, K.; and Wang, Z. 2015. Multi-exposure image fusion: A patch-wise approach. In *Proc. of ICIP*, 1717–1721.
- Ma, K.; Zeng, K.; and Wang, Z. 2015. Perceptual Quality Assessment for Multi-Exposure Image Fusion. *IEEE Transactions on Image Processing*, 24(11): 3345–3356.
- Mertens, T.; Kautz, J.; and Reeth, F. V. 2007. Exposure Fusion. In *Proc. of PG*, 382–390.
- Piella, G.; and Heijmans, H. 2003. A new quality metric for image fusion. In *Proc. of ICIP*, 173–176.
- Prabhakar, K. R.; Srikanth, V. S.; and Babu, R. V. 2017. Deepfuse: A deep unsupervised approach for exposure fusion with extreme exposure image pairs. In *Proc. of ICCV*, 4714–4722.
- Qu, L.; Liu, S.; Wang, M.; and Song, Z. 2022. Transmef: A transformer-based multi-exposure image fusion framework using self-supervised multi-task learning. In *Proc. of AAAI*, volume 36, 2126–2134.
- Rajalingam, B.; and Priya, R. 2018. Hybrid multimodality medical image fusion technique for feature enhancement in medical diagnosis. *International Journal of Engineering Science Invention*, 2(Special issue): 52–60.
- Roberts, J. W.; van Aardt, J.; and Ahmed, F. 2008. Assessment of image fusion procedures using entropy, image quality, and multispectral classification. *Journal of Applied Remote Sensing*, 2(1): 023522:1–023522:28.
- Wang, J.; Wang, W.; Xu, G.; and Liu, H. 2018. End-to-end exposure fusion using convolutional neural network. *IEICE Transactions on Information and Systems*, 101-D(2): 560–563.

- Wang, N.; and Yeung, D. 2014. Ensemble-Based Tracking: Aggregating Crowdsourced Structured Time Series Data. In *Proc. of ICML*, volume 32, 1107–1115.
- Xu, H.; Ma, J.; Jiang, J.; Guo, X.; and Ling, H. 2020a. U2Fusion: A unified unsupervised image fusion network. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(1): 502–518.
- Xu, H.; Ma, J.; Le, Z.; Jiang, J.; and Guo, X. 2020b. FusionDN: A Unified Densely Connected Network for Image Fusion. In *Proc. of AAAI*, 12484–12491.
- Xu, H.; Ma, J.; and Zhang, X.-P. 2020. MEF-GAN: Multi-exposure image fusion via generative adversarial networks. *IEEE Transactions on Image Processing*, 29: 7203–7216.
- Xydeas, C. S.; and Petrovic, V. 2000. Objective image fusion performance measure. *Electronics letters*, 36(4): 308–309.
- Zhang, H.; Xu, H.; Xiao, Y.; Guo, X.; and Ma, J. 2020a. Rethinking the image fusion: A fast unified image fusion network based on proportional maintenance of gradient and intensity. In *Proc. of AAAI*, volume 34, 12797–12804.
- Zhang, X. 2021. Benchmarking and comparing multi-exposure image fusion algorithms. *Information Fusion*, 74: 111–131.
- Zhang, Y.; Liu, Y.; Sun, P.; Yan, H.; Zhao, X.; and Zhang, L. 2020b. IFCNN: A general image fusion framework based on convolutional neural network. *Information Fusion*, 54: 99–118.
- Zhao, J.; Laganriere, R.; and Liu, Z. 2007. Performance assessment of combinative pixel-level image fusion based on an absolute feature measurement. *International Journal of Innovative Computing Information and Control*, 3(6): 1433–1447.