An empirical evaluation of convolutional neural networks for image enhancement

Pierluca D'Oro
Politecnico di Milano
pierluca.doro@mail.polimi.it

Ennio Nasca
Politecnico di Milano
ennio.nasca@mail.polimi.it

Abstract

Automatic image enhancement is an extremely relevant problem in a world where the number of produced pictures and the need for high-quality post-production modifications are enormously increasing. Convolutional neural networks have open the possibility to get to a level similar to the one of human experts. However, there is still no clear decision about the best setting for training these models in terms of both network architecture and loss function. Moreover, human experts directly leverage contextual and semantic knowledge about a picture when enhancing it. In this work, we empirically evaluate common architectures and loss functions employed for automatic image enhancement, and propose an effective architecture-agnostic method for integrating additional contextual information into the enhancement process. We evaluate the method on the MIT-Adobe Fivek dataset and show its benefits.

1. Introduction

In the last years the number of pictures taken every day in the world has grown dramatically, mostly because of easily accessible mobile cameras present in modern smartphones. Users and companies have rising desire to enhance the quality of their photos of interest. However, proper use of the tools that are included in common post-production software requires considerable expertise, and the number of individuals possessing it is not able to satisfy the outstandingly increasing demand.

While even complex combinations of classical enhancement algorithms — such as histogram equalization — lack the generality of the transformations that experts can perform, recent progress in approaches based on convolutional neural networks [16, 17] paved the way to imitate their end-to-end enhancing process. Nonetheless, many problems are still unsolved.

Due to the existence of several architectures potentially able to excel at the task, and of multiple loss functions to be chosen for optimization during the learning procedure, there is no consensus on which the best approach could be. Recent efforts [8, 6] showed approaches based on generative adversarial networks [12] can yield good results; however, training of such models can have stability issues, and can potentially obscure the actual difference in performance among different architectural choices.

Another open issue is the proper integration of side or semantic information about a picture into the end-to-end process that is trying to enhance it. Human experts unconsciously extract this knowledge: for instance, they detect which is the lighting condition or the subject of a picture while they plan how to improve it. This and other information is often available for existing pictures (e.g., in the form of tags for data that have been uploaded online) or can be easily produced using existing methods for semantic segmentation or image classification.

The main contributions of this work are:

- An empirical evaluation of state-of-the-art loss functions and architectures for deep learning-based image enhancement.
- A simple and *architecture-agnostic* method for integrating information such as tags and scene parses directly into the enhancement process.

2. Related work

The task of image enhancement using convolutional neural networks has been addressed in multiple ways. In this work, we consider a specific enhancement class, the one of enhancements produced by experts manually modifying a picture with a software. However, more classical forms of enhancement exist, such as denoising or super-resolution [7].

Deep image enhancement. [37] and [5] demonstrated that modern convolutional architectures trained with proper loss functions can approximate complex image processing operators and speed up their computation. Other work [27] uses a learned measure of image quality as loss function for tuning operators. In [13], a combination of adversarial and non-adversarial losses is used for transforming low-profile camera pictures into their high-end counterpart.

Model conditioning. Attempts to integrate semantic information into the enhancement process have been of various types. [35] and [22] employ, respectively, manually defined features and convolutional neural networks to integrate scene parsing into the learning process; [29] makes use of semantic maps for the related tasks of harmonization of image composition. Our approach to model conditioning is different to the one used by previous work on image enhancement, and is instead similar to the one of [23], that achieved significant results in tasks such as conditional image generation [2, 18] and style transfer [10].

3. Deep Image Enhancement

We frame image enhancement as a supervised learning problem. Given an image x and its enhanced version y, we are interested in obtaining an estimate $\hat{y} = \mathcal{E}(x; \theta)$ that matches the reference enhanced image as much as possible. $\mathcal{E}$ is a learned enhancement operator of parameters θ, that we model using different classes of deep convolutional neural networks. $\mathcal{E}$ can also use some *contextual information* c as additional input and compute the estimation as $\hat{y} = \mathcal{E}(x, c; \theta)$.

This is of course a restrictive view of the general enhancement task: multiple, equivalently satisfying ways exist for enhancing a single image. Model classes that take explicitly into account this multimodality [1, 26] can be a proper choice, but they are usually more impractical to optimize. We find the supervised objective to be a surprisingly good mirror to learn image enhancement: even if it is explicitly encouraged to imitate observed transformations, models supervisedly trained on enough high-quality data usually learn flexible and general enhancement mappings.

3.1. Architectures

A convolutional neural network (CNN) is an artificial neural network that employs convolution in place of affine transformations before the application of a nonlinearity. The activation a_i^l for the i-th channel of the l-th layer of a modern CNN is typically of the form:

$$a_i^l = h(\text{BN}_i^l(b_i^l + \sum_j a_j^{l-1} * K_{i,j}^l)) \tag{1}$$

where $K_{i,j}^l$ is a parameterized kernel to be convoluted with the j-th activation generated by the previous layer, b_i^l is a bias term and h is a nonlinear activation function, typically a Rectifier Linear Unit [21] or a variation of it [34]. BN is the batch normalizaiton [14] operation, known for alleviating internal covariate shift and computed on the pre-activations $\hat{a}$ of a layer as

$$\text{BN}_i^l(\hat{a}_i^l) = \gamma_i^l z + \beta_i^l \tag{2}$$

considering the z-scores z_i^l of the preactivations in a batch. At inference time, running averages of mean and standard deviation, kept during training, are used for computing the required z-scores. In this work, we considered and evaluated two standard architectures for image-to-image dense predictions tasks. In the context of image enhancement, global information is particularly useful. Hence, architectures must feature a sufficiently large receptive field, defined as the portion of input that influences the activation of a layer.

Unet. Unets [25] were conceived for image segmentation and achieve a large receptive field by using an encoder-decoder architecture. The encoder part of the model computes an increasingly small representation of the input, downsampling it through a sequence of convolutional and max-pooling [20] layers; after a bottleneck, a decoder progressively brings the resolution of the activations to the one of the input, by using a form of upsampling (e.g., bilinear) and convolution. To counterbalance the loss of information due to the use of max-pooling in the encoder, Unets make extensive use of *skip-connections*, directly offering activations computed in the encoder as input to the layers of the decoder. In practice, feature maps coming from the down-stream and the up-stream of the model are combined by channel-wise concatenation.

CAN. Context Aggregation Networks (CANs) [36], originally developed for semantic segmentation, make use of a different approach for enlarging their receptive field. The activations computed at any layer share the same resolution as the original input and the output. The context (i.e., receptive field) considered in each layer is aggregated at an exponentially increasing size with respect to depth. To achieve this, intermediate layers employ non-unitary dilation for the convolution operation. Namely, dilated convolution with dilation d is defined as:

$$a_j^{l-1} * K_{i,j}^l(t) = \sum_\tau a_j^{l-1}(t - d\tau) K_{i,j}^l(\tau) \tag{3}$$

Dilation is increased exponentially such that, after a block of dilated convolutions is started, dilation at depth l will be $d = 2^l$. Dilation is unitary for first and last layers. The number of channels of the convolutional layers is kept constant to a number of 32 (CAN32), for the entire neural network, and reduced to the same as the input with a 1×1 convolution at the output layer. To keep the resolution of the intermediate outputs of the network the same as the input, appropriate zero-padding is added as a function of dilation and of the kernel size fixed at a value of 3×3.

3.2. Loss functions

The loss that is used to optimize the model plays a crucial role in image enhancement. A multitude of loss functions have been considered in previous work and we compared six of them in order to better understand their difference in relative performance.

Typical choices are the pixel-wise *Mean Squared Error* (**MSE**) and *Mean Absolute Error* (**MAE**). The former is defined as:

$$\mathcal{L}_{\text{MSE}}(\hat{y}, y) = \frac{1}{N} \sum_i (\hat{y}_i - y_i)^2 \tag{4}$$

where N is the number of pixels in the two images. The latter is instead defined as:

$$\mathcal{L}_{\text{MAE}}(\hat{y}, y) = \frac{1}{N} \sum_i |\hat{y}_i - y_i| \tag{5}$$

Although the MSE is a standard choice in many settings for statistical prediction and machine learning, its use is often discouraged for dense prediction tasks such as image-to-image translation or image enhancement [31]. It is, in fact, more tolerant to small errors and can easily yield blurry images. The MAE is instead more appropriate to this family of tasks, and can induce the production of shrper images.

Both the previously presented losses completely ignore the perceived image quality of the estimated image, the pivotal motivation behind the image enhancement task. Therefore, in this paper we consider other losses that explicitly consider image quality as perceived by humans.

According to Webers law [11], the human visual system is more sensitive to light and color variations in homogeneous regions. Drawing from this fact, the structural similarity **SSIM** index [32] considers the structural information of a reference image as carried by three components: *luminance*, *contrast* and *structure*. These quantities are computed across local patches of size N over the estimated and ground truth images, and then compared and combined to obtain a similarity measure. A multiscale version of SSIM (MS-SSIM) [33] simulates different spatial resolutions by weighting the values of the SSIM components at different scale. A simplified version of the index considered in this work takes the form of:

$$\text{SSIM}(\hat{y}, y) = \frac{(2\mu_{\hat{y}}\, \mu_y + C_1)(2\sigma_{\hat{y}y} + C_2)}{(\mu_{\hat{y}}^2 + \mu_y^2 + C_1)(\sigma_{\hat{y}}^2 + \sigma_y^2 + C2)} \tag{6}$$

where $\hat{y}$ and y are the original and reference image signal respectively and $\text{SSIM}(\hat{y}, y) \leq 1$. The luminance μ for a generic image x is the mean intensity of the pixel values in the patch:

$$\mu_x = \frac{1}{N} \sum_{i=1}^{N} x_i \tag{7}$$

while the contrast σ is estimated as the standard deviation:

$$\sigma_x = \left[\frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)^2 \right]^{\frac{1}{2}} \tag{8}$$

Since the index is bounded the loss is computed as:

$$\mathcal{L}_{\text{SSIM}}(\hat{y}, y) = 1 - \text{SSIM}(\hat{y}, y) \tag{9}$$

Another loss function that takes into account perceptual considerations is the one proposed by [27]. It is made up of two terms, $f(\cdot)$ and $q(\cdot)$, respectively measuring fidelity to the input image and absolute quality of the enhanced image:

$$\mathcal{L}_{\text{NIMA}} = f(y, \hat{y}) + \gamma q(\hat{y}) \tag{10}$$

γ is the relative importance of the quality term. q is computed using neural image assessment **NIMA** [28], a no-reference quality index. It relies on a CNN trained on the AVA dataset [19], to learn aesthetic preferences of human raters, predicting quality ratings for images on a scale from 0 to 10. The quality term in the loss is then computed as $q = 10 - \text{NIMA}(\mathcal{E}(x))$. The fidelity function f is instead implemented as a pixel-wise loss such as $\mathcal{L}_{\text{MSE}}$ or $\mathcal{L}_{\text{MAE}}$.

3.3. Conditioning by feature-wise modulation

Feature-wise modulations [9] are a general way to introduce additional information, or *condition*, into the computational flow of an artificial neural network. The idea behind these techniques is to modify a representation by using a parameterized affine transformation, whose parameters depend on the condition to be introduced into the network. The particular form of feature-wise modulation we employ is *conditional batch normalization*, consisting in the application of the linear modulation [23] on the z-scores obtained by normalization of pre-activations:

$$\text{CBN}_i^l(\hat{a}_i^l) = \gamma_i^l(c)z + \beta_i^l(c) \tag{11}$$

The condition c can in this way up or down-regulate some of the feature maps, depending on the value of additional continuous or categorical (e.g., classes) features. In the presence of categorical features, the transformations γ and β are effectively *embeddings* from discrete to continuous values. In this work, we consider the case in which multiple categorical features are available: we obtain γ and β as a concatenation of the properly sized embeddings of the different features. If, for instance, we have 3 categorical features, the sizes of the embeddings will be such that $[\gamma_1; \gamma_2; \gamma_3]$ has length equal to the number of channels used in the convolutional layer.

This type of model conditioning is *architecture-agnostic*: we simply substitute the standard batch normalization used in intermediate layer of Unet and CAN32 with the correspondent conditional batch normalization, and feed the categorical features c as additional input to the network.

4. Experiments

We performed two sets of experiments, as presented in the introduction of the paper.

The first set had the objective to measure the relative performance of two architectures — Unet and CAN32 — and six loss functions. The loss functions we investigated are $\mathcal{L}_{\text{MSE}}$, $\mathcal{L}_{\text{MAE}}$, $\mathcal{L}_{\text{NIMA}}$ (with either $\mathcal{L}_{\text{MAE}}$ or $\mathcal{L}_{\text{MSE}}$ as fidelity function) and a uniform combination of $\mathcal{L}_{\text{MAE}}$ with $\mathcal{L}_{\text{SSIM}}$. We additionally considered the $\mathcal{L}_{\text{COLOR}}$ from [13], consisting in the Euclidean distance between the Gaussian smoothed versions of target and estimated images, and used a combination of $\mathcal{L}_{\text{SSIM}} + \mu\mathcal{L}_{\text{COLOR}}$ with $\mu = 0.00001$.

In the second set of experiments we tested whether the addition of contextual information by conditional batch normalization is beneficial for the task of image enhancement. Thus, we simply substituted all batch normalization operations with an equivalent conditional batch normalizations in the upsampling branch of the UNet and in all the intermediate layers of CAN32, and provided, for each image, side categorical features as input. We employed $\mathcal{L}_{\text{MAE+SSIM}}$ as a loss function, being it one of the most promising according to the first set of experiments.

Dataset. We used the MIT-Adobe FiveK Dataset [3] for training all the models. The dataset features 5000 professional photographs in RAW format, taken from several DSLR cameras and featuring a diverse set of scenes, subjects, and lighting conditions. For each image in the dataset, five tone-adjusted versions modified by trained experts are provided. Moreover, information about subject, light, location and time of the day in which the picture was taken is available in the form of categorical features, that we integrated into the training process as highlighted in previous sections.

We employed PNG versions of the original RAW images and choose modifications executed by the third expert as ground truth. Although the models we employed are fully convolutional and can thus handle multiple input sizes, we scaled the images of the dataset to facilitate the use of batching during training. We used a resolution of 332×500 and 500×332 for portrait and landscape images respectively and fed batches from the two categories in random order during training. We reserved 20% of the available images for testing and evaluation.

The data has been normalized in the range $[-1, 1]$ for all training and evaluation procedures except for specific pretrained models that required other normalization schemes (i.e., NIMA).

Experiments setup. For all the experiments, we employed standard Unet and CAN32. The version of CAN32 has fewer layers than the one used in other work [5], since we need a smaller receptive field for our lower resolution images. We used the Unet as presented in [25], with the sole addition of batch normalization before any activation. The exact architecture we used for CAN32 is shown in Table 2. We adopted ReLU and LeakyReLU with $\alpha = 0.2$ as nonlinear activation functions for layers of Unet and CAN32 respectively.

Optimization of all models was done using Adam [15] with $\beta_1 = 0.9$, $\beta_2 = 0.999$ and a learning rate of 0.0002. For $\mathcal{L}_{\text{NIMA}}$ we adopted $\gamma = 0.001$. We trained all models, both conditioned and unconditioned, for 50 epochs with a batch size of 8.

Results and discussion. Results of the evaluation carried out on the test data for different combinations of losses and models are shown in Table 1. Although there is no huge prevalence of an architecture over the other, the UNet performs generally better at minimizing the prescribed the loss, and the observed average quality is better than the one obtained by CAN models. Nonetheless, the UNet suffers from some pathological conditions induced by particularly bright spots in images, as can be observed in Figure 1. An advantage of the UNet architecture concerns training time: we observed that, despite a number of parameters of more than one million, compared to about 50K of CAN32, there is a significant advantage — a proportion of about 1:5 — in required training time per batch. This is mostly caused by the nature of dilated convolution.

Concerning losses, we found models trained by using $\mathcal{L}_{\text{SSIM}}$ and its variations produce outputs that are particularly pleasing from an aesthetic point of view and more resilient to artifacts.

Given the second set of experiments, we observed that, given same architecture, loss function and hyperparameters, contextual information used by conditioned models is often beneficial, especially in the case of outdoor images, as shown in Figure 2 and Figure 3. Moreover, adding this additional information consistently alleviates the bright-spot problem of UNet models.

5. Conclusion

In this work, we evaluated a number of architectures and loss functions for the task of supervised image enhancement using convolutional neural networks. We highlighted some advantages and drawbacks of the different combinations and reported some of the problems affecting existing architectures. We proposed a solution for integrating external contextual features into the enhancement process through simple layer-wise modulation, improving the results and alleviating some existing problems.

Future work can strive towards the open problem of quantitative evaluation of visual quality, relevant for both

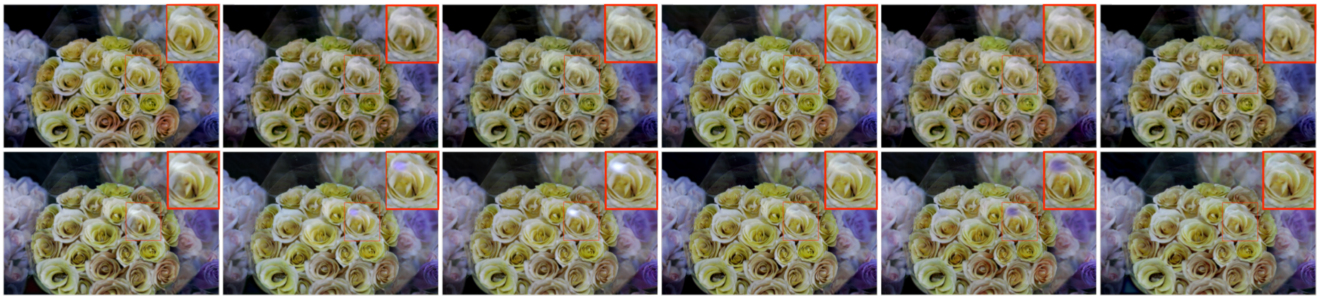

Figure 1. Image enhanced using all loss functions and models. Models from top to bottom: CAN32 and UNet. Loss functions from left to right: $\mathcal{L}_{\text{Color+SSIM}}$, $\mathcal{L}_{\text{MAE}}$, $\mathcal{L}_{\text{MAE+NIMA}}$, $\mathcal{L}_{\text{MAE+SSIM}}$, $\mathcal{L}_{\text{MSE}}$ and $\mathcal{L}_{\text{MSE+NIMA}}$.

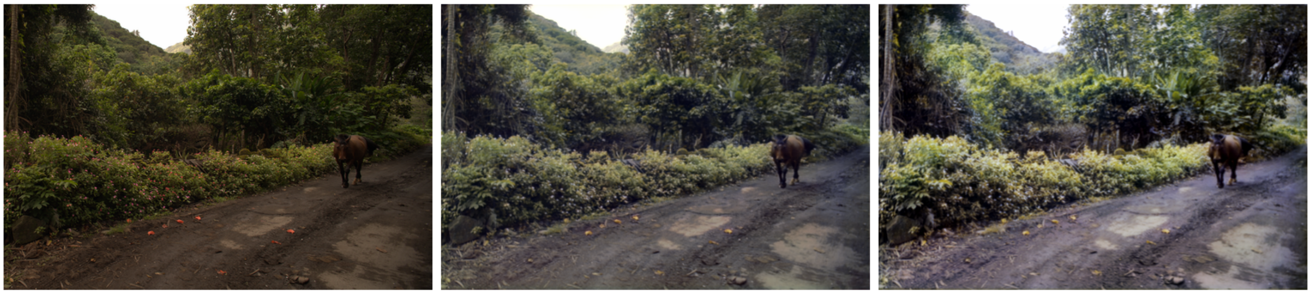

Figure 2. Results obtained using a $\mathcal{L}_{\text{MAE+SSIM}}$ loss function. From left to right: Original image, Output from CAN32, Output from CAN32 conditioned by contextual information. The conditioned model performs better especially in outdoor images, as shown by the improved color of the subject and the vegetation.

image enhancement and related tasks such as image generation. We carried out a survey about preferences over images generated by the different models we presented, but we obtained inconclusive results, mainly due to the difficulty of evaluating a total of 12 architecture/loss combinations at the same time. Another interesting direction is the extension of this work to the use of semantic segmentation information. We carried out some experiments on the use of maps generated by existing networks trained for semantic segmentation, leveraging the same simple conditioning mechanism we used with contextual features, but did not reach satisfying results. We think this kind of side information can be particularly beneficial in the case of adversarial training, as shown, for instance, in recent approaches presented for the task of video generation [4, 30].

A. Experiments on learning adaptive histogram equalization

A preliminary experiment to gain some insight about image enhancement was carried out on the CIFAR10 dataset. The task consisted in learning a traditional image-to-image transformation. In particular, we focused on a variation of adaptive histogram equalization called CLAHE [24], which performs histogram equalization over patches, putting a limit on contrast to avoid noise amplification. We trained three simple architectures using $\mathcal{L}_{\text{MSE}}$, namely a Multilayer perceptron working on the stretched image, a LeNet-like [17] convolutional neural network and a small UNet with skip connections.

Although the very small resolution (32×32) of the images in CIFAR10 did not allow any meaningful visual inspection, we found that, even in such a simple task, the skip-connections employed by the UNet, that reports the best performance on the test set in terms of $\mathcal{L}_{\text{MSE}}$, are extremely beneficial. This offers an hint on why this kind of architectures are widely employed for learning image-to-image transformations.

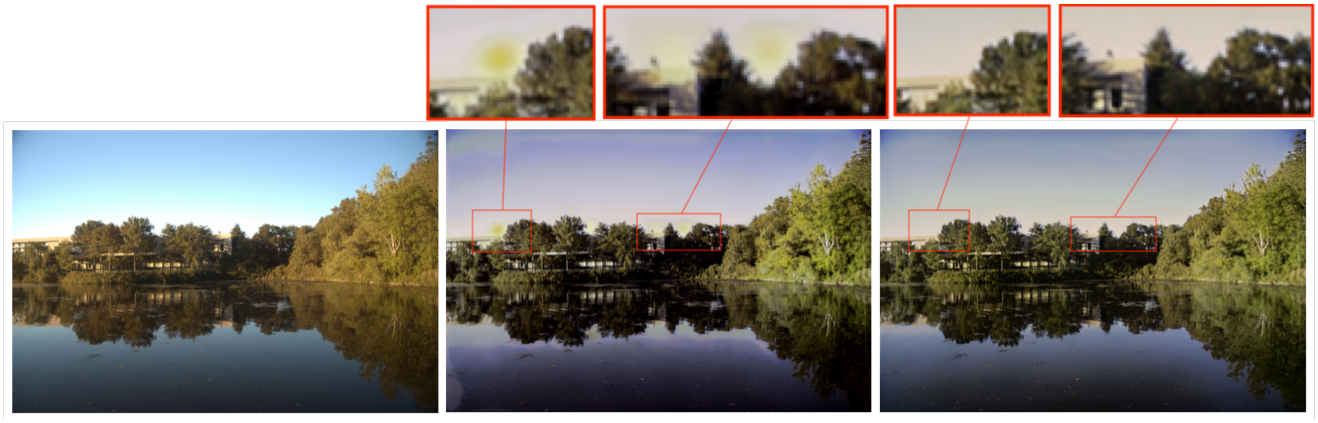

Figure 3. Results obtained using a $\mathcal{L}_{\text{MAE+SSIM}}$ loss function. From left to right: Original image, Output from Unet, Output from Unet conditioned by contextual information. In the highlighted areas, it is shown that the conditioned model is able to correctly enhance particularly bright areas, that instead induce artifacts for the unconditioned model.

Model	$\mathcal{L}_{\text{MAE}}$	$\mathcal{L}_{\text{MSE}}$	$\mathcal{L}_{\text{MAE+SSIM}}$	$\mathcal{L}_{\text{COLOR+SSIM}}$	$\mathcal{L}_{\text{MAE+NIMA}}$	$\mathcal{L}_{\text{MSE+NIMA}}$
CAN32	**0.1414**	0.0463	0.5102	0.4530	**0.0443**	0.0451
UNet	0.1421	**0.0364**	**0.4676**	**0.4239**	0.1429	**0.0414**

Table 1. Results obtained for the analyzed model-loss pairs.

Layer	1	2	3	4	5	6	7	8
Convolution	3× 3	3× 3	3× 3	3× 3	3× 3	3× 3	3× 3	1× 1
Dilation	1	2	4	8	16	32	1	1
Receptive Field	3× 3	7× 7	15× 15	31× 31	63× 63	127× 127	129× 129	129× 129
Nonlinearity	✓	✓	✓	✓	✓	✓	✓	✗
Channels	32	32	32	32	32	32	32	3

Table 2. Specification for the CAN model.

References

[1] C. M. Bishop. Mixture density networks. Technical report, 1994. 2

[2] A. Brock, J. Donahue, and K. Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. 2

[3] V. Bychkovsky, S. Paris, E. Chan, and F. Durand. Learning photographic global tonal adjustment with a database of input / output image pairs. In *The Twenty-Fourth IEEE Conference on Computer Vision and Pattern Recognition*, 2011. 4

[4] C. Chan, S. Ginosar, T. Zhou, and A. A. Efros. Everybody dance now. *CoRR*, abs/1808.07371, 2018. 5

[5] Q. Chen, J. Xu, and V. Koltun. Fast image processing with fully-convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2497–2506, 2017. 1, 4

[6] Y. C. Chuang. Deep photo enhancer: Unpaired learning for image enhancement from photographs with gans. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6306–6314, 2018. 1

[7] R. Dahl, M. Norouzi, and J. Shlens. Pixel recursive super resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 5439–5448, 2017. 1

[8] Y. Deng, C. C. Loy, and X. Tang. Aesthetic-driven image enhancement by adversarial learning. In *ACM Multimedia*, 2018. 1

[9] V. Dumoulin, E. Perez, N. Schucher, F. Strub, H. d. Vries, A. Courville, and Y. Bengio. Feature-wise transformations. *Distill*, 2018. https://distill.pub/2018/feature-wise-transformations. 3

[10] V. Dumoulin, J. Shlens, and M. Kudlur. A learned representation for artistic style. *CoRR*, abs/1610.07629, 2016. 2

[11] G. Ekman. Weber's law and related functions. *The Journal of Psychology*, 47(2):343–352, 1959. 3

[12] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014. 1

[13] A. Ignatov, N. Kobyshev, R. Timofte, K. Vanhoey, and L. V. Gool. Dslr-quality photos on mobile devices with deep convolutional networks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 3297–3305, 2017. 2, 4

[14] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015. 2

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 4

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Commun. ACM*, 60:84–90, 2012. 1

[17] Y. LeCun, L. Bottou, and P. Haffner. Gradient-based learning applied to document recognition. 2001. 1, 5

[18] T. Miyato and M. Koyama. cgans with projection discriminator. *CoRR*, abs/1802.05637, 2018. 2

[19] N. Murray, L. Marchesotti, and F. Perronnin. Ava: A large-scale database for aesthetic visual analysis. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2408–2415. IEEE, 2012. 3

[20] J. Nagi, F. Ducatelle, G. A. D. Caro, D. C. Ciresan, U. Meier, A. Giusti, F. Nagi, J. Schmidhuber, and L. M. Gambardella. Max-pooling convolutional neural networks for vision-based hand gesture recognition. *2011 IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pages 342–347, 2011. 2

[21] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th international conference on machine learning (ICML-10)*, pages 807–814, 2010. 2

[22] S. Nam and S. J. Kim. Deep semantics-aware photo adjustment. *CoRR*, abs/1706.08260, 2017. 2

[23] E. Perez, F. Strub, H. de Vries, V. Dumoulin, and A. C. Courville. Film: Visual reasoning with a general conditioning layer. In *AAAI*, 2018. 2, 3

[24] S. M. Pizer, E. P. Amburn, J. D. Austin, R. Cromartie, A. Geselowitz, T. Greer, B. ter Haar Romeny, J. B. Zimmerman, and K. Zuiderveld. Adaptive histogram equalization and its variations. *Computer vision, graphics, and image processing*, 39(3):355–368, 1987. 5

[25] O. Ronneberger, P. Fischer, and T. 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. 2, 4

[26] R. Salakhutdinov. Learning deep generative models. *Annual Review of Statistics and Its Application*, 2:361–385, 2015. 2

[27] H. Talebi and P. Milanfar. Learned perceptual image enhancement. In *2018 IEEE International Conference on Computational Photography (ICCP)*, pages 1–13. IEEE, 2018. 1, 3

[28] H. Talebi and P. Milanfar. Nima: Neural image assessment. *IEEE Transactions on Image Processing*, 27(8):3998–4011, 2018. 3

[29] Y.-H. Tsai, X. Shen, Z. L. Lin, K. Sunkavalli, X. Lu, and M.-H. Yang. Deep image harmonization. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2799–2807, 2017. 2

[30] T.-C. Wang, M.-Y. Liu, J.-Y. Zhu, G. Liu, A. Tao, J. Kautz, and B. Catanzaro. Video-to-video synthesis. In *NeurIPS*, 2018. 5

[31] Z. Wang and A. C. Bovik. Mean squared error: Love it or leave it? a new look at signal fidelity measures. *IEEE Signal Processing Magazine*, 26:98–117, 2009. 3

[32] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 3

[33] Z. Wang, E. P. Simoncelli, and A. C. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003. 3

[34] B. Xu, N. Wang, T. Chen, and M. Li. Empirical evaluation of rectified activations in convolutional network. *CoRR*, abs/1505.00853, 2015. 2

[35] Z. Yan, H. Zhang, B. Wang, S. Paris, and Y. Yu. Automatic photo adjustment using deep neural networks. *ACM Trans. Graph.*, 35:11:1–11:15, 2016. 2

[36] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016. 2

[37] H. Zhao, O. Gallo, I. Frosio, and J. Kautz. Loss functions for neural networks for image processing. *CoRR*, abs/1511.08861, 2015. 1