This CVPR 2020 workshop paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Learned Low Bit-rate Image Compression with Adversarial Mechanism

Jiayu Yang*, Chunhui Yang*, Yi Ma*, Shiyi Liu, Ronggang Wang
Peking University, China
{jiayuyang, yangchunhui, mayi, shy_liu11}@pku.edu.cn {rgwang}@pkusz.edu.cn

Abstract

Adversarial mechanism is introduced to learned image compression system in this paper. Our motivation is that the number of quantization levels is limited with the constraint of low bit-rate, resulting in severe distortion in details after reconstruction. The adversarial training manner enhances the ability of Decoder/Generator to enrich textures and details in the reconstructed image. Channel-spatial attention mechanism is used to refine the intermediate features implicitly to boost the representation power of CNNs. As for entropy model, we jointly take hyperpriors and autoregressive priors for accurate probability estimation. Moreover, an EDSR-like post-processing subnetwork is concatenated after Decoder for further quality enhancement. The proposed approach demonstrates competitive performance when evaluated with multi-scale structural similarity (MS-SSIM) and favorably visual quality at low bit-rate.

1. Introduction

Deep learning has been widely applied in image compression tasks and achieves a promising performance in recent years. Many image compression works based on deep learning have been proposed, which can be roughly divided into two categories. The first kind is to use deep learning to enhance tools of traditional image compression codecs or add post-processing modules, such as the approach of Prakash *et al*. [18]. The other completely uses deep learning in an end-to-end optimized manner for image compression. In [21, 22, 15, 6], Recurrent Neural Network (RNN) is applied to image compression. In each iteration, the encoder generates a binary latent representation. By increasing the number of iterations, the bits streams become larger and the quality of the reconstructed images can be enhanced. Though the recurrent manner can naturally handle the problem of variable-rate compression, it usually takes more time in practical application. Different from these recurrent models that usually need to be executed more than once, the following methods compress images in a feed-forward manner. In [2, 20, 16], entropy models with fixed parameters are studied in the compression framework, which are optimized for the rate estimation and entropy coding to improve the effect of image compression. After that, Ballé *et al*. [3] design a hyperprior network based on the entropy model to estimate the scales of latent representations so that the bit-rate can be estimated more accurately. Lee *et al*. [7] and Minnen *et al*. [14] also propose approaches that combine autoregression with hyperprior to estimate the entropy of the latent representation, which demonstrate better performance than BPG [4]. In [8, 13], importance maps of image content are introduced , according to which the number of bits is allocated, making the important areas of the images in better reconstructed quality. Moreover, Rippel *et al*. [19] and Agustsson *et al*. [1] adopt generative adversarial model to enhance the quality of reconstructed images, which shows better subjective quality with bit-rate greatly reduced. In our framework, we leverage this capability at the decoder side and design a powerful encoder to extract saliency feature for better reconstruction quality.

In this paper, we propose an image compression framework based on variational autoencoder. A channel-spatial attention block (CSAB) is introduced as the basic block in our compression framework, guiding the convolutional neural network (CNN) to allocates more bits to salient features implicitly. Thus, the *Main Decoder* can reconstruct images in better quality with limited bit-rate constraints. In addition, we introduce an adversarial mechanism for our compression framework, which enriches details of reconstruction images and enhances the subjective quality. The adversarial loss is incorporated with rate-distortion loss, formulating a multi-task learning problem. We also introduce an EDSR-like post-processing module[9], an image enhancement network for super-resolution tasks, to further improve the quality of reconstruction. With the proposed pipeline, our image compression framework demonstrates competitive performance in terms of multi-scale structural similarity (MS-SSIM) and pleasing visual quality.

*These authors contribute equally.

†This work is supported by National Natural Science Foundation of China 61672063, Shenzhen Research Projects of JCYJ20180503182128089 and 20180608092141929O.

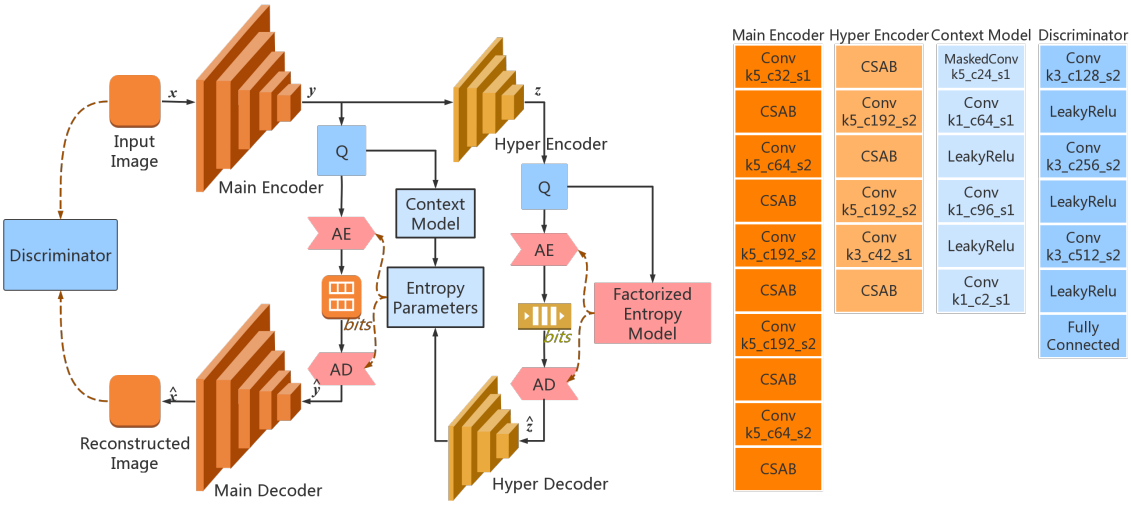


Figure 1. Network architecture of proposed model. AE and AD represent arithmetic encoder and arithmetic decoder. Q stands for *round* quantization. Details about parameter settings are shown on the right. "Conv" denotes a convolutional layer, "k" represents the kernel size, "c" denotes the number of channels and "s" is the stride. "CSAB" represents proposed channel-spatial attention block, details of which can be seen in Figure 2. *Main Decoder* and *Hyper Decoder* have a symmetrical structure with *Main Encoder* and *Hyper Encoder*, except that convolutional layers with stride 2 for down-sampling are replaced with transposed convolutional layer with stride 2 for up-sampling.

2. Approach

Figure 1 provides a high-level overview of our proposed method. An autoencoder learns a compact latent representation of input images (*Main Encoder* and *Main Decoder* blocks), followed by an entropy model for conditional probability estimation over the quantized latent representation (*Hyper Encoder*, *Hyper Decoder* and *Context Model*). Then, both reconstructed image and input image are fed into a *Discriminator* for adversarial training. Parameter settings of proposed model are demonstrated on the right of Figure1. The *Main Decoder* and *Hyper Decoder* have a symmetrical architecture with *Main Encoder* and *Hyper Encoder*.

2.1. Channel-spatial Attention Block

We introduce a channel-spatial attention block (CSAB) as the basic block of the model, which is stacked in both main and hyper autoencoder. The architecture of proposed CSAB is shown in Figure 2. Piped residual blocks maintain the network's capacity for powerful feature extraction. The batch normalization and non-linear activation function after residual connection in the residual block are removed, as that in EDSR[9], due to the fact that the decoding procedure is somewhat similar to super-resolution task since both are dense prediction tasks involving spatial upsampling [12]. GDN/IGDN is used as non-linear activation, which implements local divisive normalization transformation and is proven to be particularly suitable for density modeling and image compression[2].

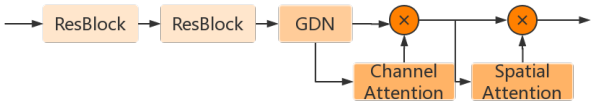


Figure 2. Architecture of channel-spatial attention block (CSAB). Channel-spatial attention module refines the intermediate features.

Inspired by the characteristic of distributed representations of representation learning [5], a simple but effective channel-spatial attention module [23] is used to refine intermediate features and allocate more bits to salient features implicitly which are critical for reconstruction. The channel attention focuses on 'what' is meaningful in the input image while the spatial attention focuses on 'where' is an informative part, thus they are complementary to boost representation power of CNNs. Let $\mathbf{F} \in \mathbb{R}^{C\times H\times W}$ denotes the intermediate feature map, channel attention module infers a 1D channel attention map $\mathbf{M_c} \in \mathbb{R}^{C\times 1\times 1}$ and spatial attention module infers a 2D spatial attention map $\mathbf{M_s} \in \mathbb{R}^{1\times H\times W}$:

$$\begin{aligned}\mathbf{M_c}(\mathbf{F}) = \sigma(\mathbf{W_1}(\mathbf{W_0}(AvgPool(\mathbf{F}))) \\ +\mathbf{W_1}(\mathbf{W_0}(MaxPool(\mathbf{F}))))\end{aligned} \quad (1)$$

$$\mathbf{M_s}(\mathbf{F}) = \sigma\left(f^{5\times 5}([AvgPool(\mathbf{F}); MaxPool(\mathbf{F})])\right) \quad (2)$$

where σ denotes the sigmoid function, $\mathbf{W_0}$ and $\mathbf{W_1}$ are shared weights for different input. *AvgPool* and *MaxPool* stand for *Average Pooling* and *Max Pooling* operations. $f^{5\times 5}$ represents a convolution layer with kernel size of 5×5.

Finally, the channel attention maps $\mathbf{M_c}$ and spatial attention maps $\mathbf{M_s}$ refine $\mathbf{F}$ by element-wise multiplication:

$$\begin{aligned}\mathbf{F}' &= \mathbf{M_c}(\mathbf{F}) \otimes \mathbf{F} \\ \mathbf{F}'' &= \mathbf{M_s}(\mathbf{F}') \otimes \mathbf{F}'\end{aligned} \tag{3}$$

2.2. Enriching Details by Adversarial Training

The range of quantization levels for latent representations learned by *Encoder* are limited under the constraints of low bit-rate, making the reconstructed images hardly restore details and suffer from strong distortion. Hence, an adversarial training manner is introduced to fill the gap between the reconstructed image and the input image. Specifically, the reconstructed image $\hat{x}$ and input image x are fed into a *Discriminator*, yielding powerful generator (*Decoder*) which captures both local texture and global semantic information under the guidance of adversarial loss.

The adversarial training manner is formulated as a binary classification problem, *i.e.*, our adversarial loss operates on classifying the 'real one' from pairs of real/fake images, as that in [19], and is formulated as,

$$L_D = \frac{1}{N}\sum_{n=1}^{N}\left(L_{BCE}\left(x,1\right) + L_{BCE}\left(\hat{x},0\right)\right) \tag{4}$$

$$L_G = \frac{1}{N}\sum_{n=1}^{N} L_{BCE}\left(\hat{x},1\right) \tag{5}$$

where L_{BCE} is binary cross entropy loss.

2.3. Quantization

The low-dimension representation of image, *i.e.*, latent representations, shall be quantized then coded. Usually we use the *round* function for quantization. However, the quantization leads to zero gradient almost everywhere, making it ineffective to train the network via gradient descent. Following the work of Ballé *et al.* [2], we replace the quantizer with additive i.i.d uniform noise during training:

$$\hat{y}_i = y_i + noise \sim U(-\frac{1}{2},\frac{1}{2}) \tag{6}$$

where $\hat{y}_i$ represents elements of quantized latent features.

2.4. Conditional Probability Estimation

We jointly leverage autoregressive priors and hyperpriors for probability estimation by concatenating features from *Context Model* and *Hyper Decoder*. The architecture of *Context Model* for autoregressive priors is shown in Figure 1. Inspired by the idea of PixelCNN[17], we predict the current pixel by leveraging the neighboring decoded pixels to make full use of the spatial and cross-channel correlation, which is implemented by a 3D masked convolution. Besides, a parallel manner for 3D masked convolution[11] is used to further accelerate the predicting procedure. Following Minnen *et al.* [14], we model the distribution of each element $\hat{y}_i$ in quantized latent features $\hat{y}$ as a conditional Gaussian distribution with mean value μ_i and standard deviation σ_i:

$$\begin{aligned}p_{\hat{y}}(\hat{y}_i|\hat{y}_1,...,\hat{y}_{i-1},\hat{z}) = \\ \prod_i (\mathcal{N}(\mu_i,{\sigma_i}^2) * \mathcal{U}(-\frac{1}{2},\frac{1}{2}))(\hat{y}_i)\end{aligned} \tag{7}$$

where μ_i and σ_i are predicted conditioned on hyperprior $\hat{z}$ and causal (and possibly reconstructed) pixels prior to $\hat{y}_i$. The casual context is denoted as $\hat{y}_1, \hat{y}_2, ..., \hat{y}_{i-1}$.

Hyperpriors $\hat{z}$, which is used to capture the spatial dependencies of latent representations [3], can be modeled by a non-parametric, fully factorized density model:

$$p_{\hat{z}|\psi}(\hat{z}|\psi) = \prod_i (p_{z_i|\psi^{(i)}}(\psi^{(i)}) * \mathcal{U}(-\frac{1}{2},\frac{1}{2}))(\hat{z}_i) \tag{8}$$

where $\psi^{(i)}$ represents the parameters of each univariate distribution $p_{\hat{z}|\psi^{(i)}}$. Therefore, bit rate of $\hat{y}$ and $\hat{z}$ can be evaluated as:

$$R_{\hat{y}} = -\sum\nolimits_i \log_2(p_{\hat{y}_i|\hat{z}_i}(\hat{y}_i|\hat{z}_i)) \tag{9}$$

$$R_{\hat{z}} = -\sum\nolimits_i \log_2(p_{\hat{z}_i|\psi^{(i)}}(\hat{z}_i|\psi^{(i)})) \tag{10}$$

2.5. Multi-task Learning

We introduce adversarial loss to the general Rate-Distortion optimization, formulating a multi-task learning problem. The joint objective is to minimize the combination of the distortion loss, rate loss as well as adversarial loss with λ_1 and λ_2 as trade-off parameters to balance different loss. Thus, the objective function is defined as

$$L = R + \lambda_1\left(D + \lambda_2 L_G\right) \tag{11}$$

where $R = R_{\hat{y}} + R_{\hat{z}}$ denotes the rate loss, and $D = 1 - MS\text{-}SSIM$ denotes the distortion loss.

2.6. Post-processing

By observing the reconstructed images, we find that some details of the reconstructed image are blurred, so we introduce an enhanced sub-network oriented to super-resolution tasks as post-processing module to enrich the details of reconstructed images. The architecture of proposed EDSR-like post-processing sub-network composes of convolutional layers and 20 residual blocks, as is shown in Figure 3. In the sub-network, skip connection maintains the efficiency of deep networks. The batch normalization layers are removed from the residual blocks so that the post-processing network can contain more residual blocks and extract more useful features. Meanwhile, we introduce a constant scaling layer in the residual block, with which our post-processing sub-network can be trained more steadily.

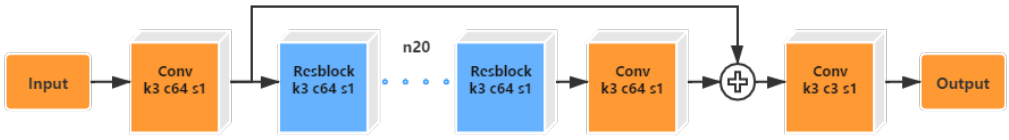
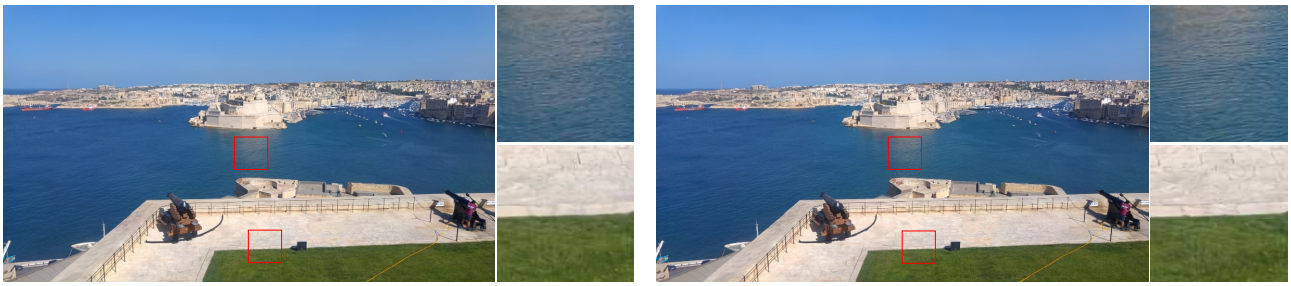


Figure 3. Architecture of post-processing sub-network, where "k" represents kernel size, "c" denotes number of filters, and "s" is stride of a convolutional layer. "n20" represents 20 residual blocks.



(a) Proposed method without adversarial mechanism, 0.169bpp

(b) Proposed method, 0.174bpp

Figure 4. Comparison on visual quality. Sampled patches are listed by the right. The adversarial manner enriches textures and details.

Table 1. Evaluation results on CLIC 2020 validation datasets.

Methods	PSNR(dB)	MS-SSIM	Bit Rate(BPP)	Decoder size(Byte)	Decoding time(s)
Proposed	29.220	0.9729	0.149	220378325	11414
W/O Post-processing	29.118	0.9725	0.149	214865170	11988
MIATLSSIM	30.170	0.9781	0.15	475395523	14508
VIP-ICT-Codec	32.625	0.9635	0.15	287490775	1703
BPG444	31.049	0.9514	0.15	377869	71
JPEG420	26.488	0.8696	0.15	208	33

3. Experiment

CLIC2020 training set and COCO dataset[10] are used as training set, in which all the images are random cropped into patches with size 256x256. All the modules in our approach are trained in an end-to-end manner with an ADAM optimizer. Different values of hyper parameter λ_1 in range [4, 8] are chosen to reach different bit rate, and λ_2 is set as $1e^{-4}$. The learning rate decreases from $1e^{-4}$ to $1e^{-6}$ by 0.1 after every 100,000 iterations.

Our results in valid phase are shown in Table 1, which achieve competitive performance in MS-SSIM with smaller decoder size compared with other learned image compression methods. *W/O Post-processing* denotes our results with post-processing sub-network removed, which proves the effectiveness of the introduced post-processing sub-network.

An ablation study on our proposed framework is shown in Figure 4 to investigate the effectiveness of adversarial mechanism. As is observed, the reconstructed image with adversarial mechanism expresses more natural textures and details, leading to better visual quality with limited bit-rate.

4. Conclusion

Adversarial loss is introduced in this paper to compensate severe distortion in details for date-driven image compression under the constraint of low bit-rate. Besides, motivated by the distributed representation characteristic of autoencoders, channel-spatial attention module is used to emphasize the salient features. Moreover, an EDSR-like post-processing sub-network enhances the quality of reconstructed image. The experiments and ablation study show the superiority of our approach in enriching details of the image, leading to pleasing visual quality.

For future work, a more efficient entropy model will be explored to reduce the bit-rate and accelerate the encoding as well as decoding procedure. Though combining hyperpriors with autoregressive priors for conditional probability estimation shows state-of-the-art performance on entropy estimation, it is time-consuming due to the sequential nature of autoregressive model.

References

[1] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool. Generative adversarial networks for extreme learned image compression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 221–231, 2019. 1

[2] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016. 1, 2, 3

[3] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston. Variational image compression with a scale hyperprior. *arXiv preprint arXiv:1802.01436*, 2018. 1, 3

[4] Fabrice Bellard. Bpg image format. *URL https://bellard.org/bpg*, 2015. 1

[5] Bengio, Yoshua, Courville, Aaron, Vincent, and Pascal. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis Machine Intelligence*, 35(8):1798–1828, 2013. 2

[6] Nick Johnston, Damien Vincent, David Minnen, Michele Covell, Saurabh Singh, Troy Chinen, Sung Jin Hwang, Joel Shor, and George Toderici. Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4385–4393, 2018. 1

[7] Jooyoung Lee, Seunghyun Cho, and Seung-Kwon Beack. Context-adaptive entropy model for end-to-end optimized image compression. *arXiv preprint arXiv:1809.10452*, 2018. 1

[8] Mu Li, Wangmeng Zuo, Shuhang Gu, Debin Zhao, and David Zhang. Learning convolutional networks for content-weighted image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3214–3223, 2018. 1

[9] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 136–144, 2017. 1, 2

[10] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 740–755, 2014. 4

[11] Haojie Liu, Tong Chen, Qiu Shen, and Zhan Ma. Practical stacked non-local attention modules for image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2019. 3

[12] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Practical full resolution learned lossless image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10629–10638, 2019. 2

[13] Fabian Mentzer, Eirikur Agustsson, Michael Tschannen, Radu Timofte, and Luc Van Gool. Conditional probability models for deep image compression. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4394–4402, 2018. 1

[14] David Minnen, Johannes Ballé, and George D Toderici. Joint autoregressive and hierarchical priors for learned image compression. In *Advances in Neural Information Processing Systems*, pages 10771–10780, 2018. 1, 3

[15] David Minnen, George Toderici, Michele Covell, Troy Chinen, Nick Johnston, Joel Shor, Sung Jin Hwang, Damien Vincent, and Saurabh Singh. Spatially adaptive image compression using a tiled deep network. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 2796–2800. IEEE, 2017. 1

[16] David Minnen, George Toderici, Saurabh Singh, Sung Jin Hwang, and Michele Covell. Image-dependent local entropy models for learned image compression. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 430–434. IEEE, 2018. 1

[17] Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks. *arXiv preprint arXiv:1601.06759*, 2016. 3

[18] Aaditya Prakash, Nick Moran, Solomon Garber, Antonella DiLillo, and James Storer. Semantic perceptual image compression using deep convolution networks. In *2017 Data Compression Conference (DCC)*, pages 250–259. IEEE, 2017. 1

[19] Oren Rippel and Lubomir Bourdev. Real-time adaptive image compression. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2922–2930. JMLR. org, 2017. 1, 3

[20] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *arXiv preprint arXiv:1703.00395*, 2017. 1

[21] George Toderici, Sean M O'Malley, Sung Jin Hwang, Damien Vincent, David Minnen, Shumeet Baluja, Michele Covell, and Rahul Sukthankar. Variable rate image compression with recurrent neural networks. *arXiv preprint arXiv:1511.06085*, 2015. 1

[22] George Toderici, Damien Vincent, Nick Johnston, Sung Jin Hwang, David Minnen, Joel Shor, and Michele Covell. Full resolution image compression with recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5306–5314, 2017. 1

[23] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 2