This CVPR 2021 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.

Deep Homography for Efficient Stereo Image Compression

Xin Deng[1], Wenzhe Yang[2], Ren Yang[3], Mai Xu[2,*], Enpeng Liu[2], Qianhan Feng[2], Radu Timofte[3]
[1]School of Cyber Science and Technology, Beihang University, Beijing, China
[2]School of Electronic and Information Engineering, Beihang University, Beijing, China
[3]Computer Vision Lab, D-ITET, ETH Zurich, Zurich, Switzerland
{cindydeng, ywzsunny}@buaa.edu.cn, ren.yang@vision.ee.ethz.ch, MaiXu@buaa.edu.cn,
{liuepbuaa, fqhank}@gmail.com, radu.timofte@vision.ee.ethz.ch

Abstract

In this paper, we propose HESIC, an end-to-end trainable deep network for stereo image compression (SIC). To fully explore the mutual information across two stereo images, we use a deep regression model to estimate the homography matrix, i.e.*, H matrix. Then, the left image is spatially transformed by the H matrix, and only the residual information between the left and right images is encoded to save bit-rates. A two-branch auto-encoder architecture is adopted in HESIC, corresponding to the left and right images, respectively. For entropy coding, we use two conditional stereo entropy models,* i.e.*, Gaussian mixture model (GMM) based and context based entropy models, to fully explore the correlation between the two images to reduce the coding bit-rates. In decoding, a cross quality enhancement module is proposed to enhance the image quality based on inverse H matrix. Experimental results show that our HESIC outperforms state-of-the-art SIC methods on InStereo2K and KITTI datasets both quantitatively and qualitatively. Code is available at* https://github.com/ywz978020607/HESIC.

1. Introduction

Stereo image compression (SIC) aims to jointly compress a pair of stereoscopic left and right images, to achieve high compression rate for both the two images. In the fields of autonomous driving [45], virtual reality [19] and video surveillance [12], SIC has become one of the most critical techniques, which recently attracts increasing attention from both academic and industrial communities. By fully exploiting the mutual information in the two images, SIC can potentially achieve higher compression rate than compressing each image independently.

Compared with single image compression [4], SIC is more challenging, which needs to fully exploit and utilize the inner relationship between the left and right stereo images. There have been many research works on traditional stereo image compression [18, 8, 27]. However, they rely on hand-crafted features and the traditional optimization theory to minimize the rate-distortion loss, which have limited compression efficiency. Recently, Liu *et al*. [32] proposed the first deep learning based stereo image compression method named DSIC, which exploits the content redundancy between the stereo pair to reduce the joint bit-rate. However, it has very high computational complexity due to the dense warp scheme. In addition, it requires that the left and right images must stand in the same horizontal line, which is actually not feasible in practical applications.

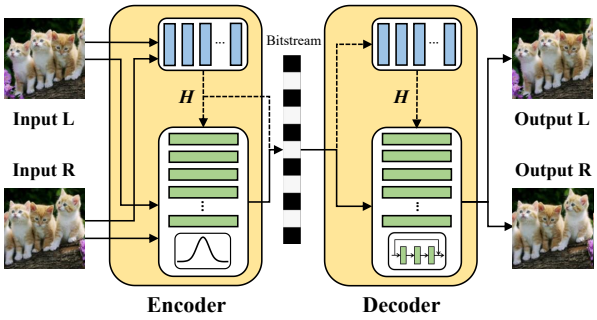


Figure 1. Brief framework of the proposed HESIC approach for stereo image compression. Here, the "H" indicates homography transformation.

In this paper, we propose an efficient stereo image compression network based on homography transformation, namely HESIC, which overcomes all the aforementioned drawbacks of DSIC. As shown in Fig. 1, we use homography transformation [35] to replace the dense warp module in DSIC [32], which can significantly decrease the computational complexity. In addition, since homography transformation has no requirement for the relative position of the two images, our method can cope with the case when the two stereo images are not in the same horizontal line. Finally, our HESIC approach not only outperforms the state-of-the-art

*Corresponding author.

deep learning based single image compression methods, but also saves around 31.7% bit-rate compared to the latest SIC method DSIC with similar image quality.

The main contributions of this paper are as follows:

- We propose a novel stereo image compression network based on homography transformation, namely HESIC, for the task of stereo image compression.
- We introduce two conditional entropy models specifically for stereo image compression, aiming to fully reduce the redundancy between two stereo images.
- We develop a cross quality enhancement module in the decoder, which is able to enhance the image quality using inverse homography matrix.

2. Related work

2.1. Single image compression

The traditional lossy image compression is typically composed of three components: transformation, quantization and entropy coding. The image is firstly transformed from pixel domain to some frequency domains, to make the energy concentrate on a few transform coefficients. These coefficients are then quantized and encoded by entropy model for transmission and storage. The joint photographic experts group (JPEG) standard is the most well-know image compression method, which applies discrete cosine transform (DCT) on image blocks, to make the energy concentrated in Fourier domain [42]. Different from JPEG, the discrete wavelet transform is adopted in JPEG2000 where the energy is concentrated in wavelet domain [37, 22]. Due to the multi-scale orthogonal wavelet decomposition, JPEG 2000 has superior compression ratios than JPEG. There are also some video compression methods which can be used for image compression. For example, the high efficient video coding (HEVC) standard provides several configurations, such as intra-frame, random access and low delay [39]. In particular, the intra-frame configuration compresses each frame independently, which can be used for image compression. The better portable graphics (BPG) [7] image compression algorithm proposed by Fabrice Bellard is developed based on the intra-frame encoding of HEVC.

The transformation, quantization and entropy coding are usually treated independently in traditional image compression methods, *i.e*., they are separately optimized. Recent years have seen some deep learning based methods which jointly optimize the three components through end-to-end training [40, 3, 2, 31, 4, 34, 23, 1, 47, 43, 16, 14, 15]. The advantage of deep neural network (DNN) is to learn the nonlinear functions to map pixels into a more compressible latent space, which can potentially improve the compression ratio. Most DNN based image compression methods adopt an encoder-decoder network architecture. The encoder aims to encode the image into latent representations, and the decoder is designed to recover the image from the latent representations. Specifically, Toderici *et al*. [40] proposed a recurrent neural network (RNN) based encoder and decoder for image compression with variable compression rates. Later, Ballé *et al*. [4] proposed an end-to-end image compression model based on variational autoencoders. To deal with the extreme low-bitrate image compression, Li *et al*. [31] proposed a deep network considering the saliency of image content at different locations. Most recently, Wang *et al*. [43] proposed an invertible encoding module to replace the conventional encoder-decoder structure, which achieves comparable image compression performance with less network parameters.

2.2. Stereo image matching and compression

Different from single image compression, SIC aims to jointly compress two stereo images by exploring the mutual information between them. The stereo image matching is the core technique in SIC, which can be broadly classified into two categories, rigid [10, 35, 17, 25] and non-rigid [36, 46, 38] matching. The non-rigid stereo matching methods provide non-uniform pixel warp function, which is more flexible but difficult to be learned, and they often need a large amount of computational resources. Compared to non-rigid matching, the rigid matching is much easier to implement. The homography perspective transformation [10, 35, 17, 25] is the typical technique for rigid matching, which is widely used in image stitching, 3D reconstruction, etc. To calculate the homography matrix, the traditional methods firstly use feature matching algorithms such as SIFT [33] and SURF [6], to get feature points, and then use RANdom SAmple Consensus (RANSAC) algorithm [13] to calculate the transformation matrix. Recently, Nguyen *et al*. [35] proposed to use unsupervised deep learning to get the homography matrix between two images.

Recently, Liu *et al*. [32] proposed a deep stereo image compression (DSIC) network, which achieves SOTA performance in SIC. This is also the only DNN based work we can find specifically for SIC. In this work, the parameteric skip functions and a conditional entropy model were proposed to model the dependence between the left and right images, leading to significant bit-rate saving in SIC. However, DSIC still has some disadvantages. Firstly, it has high computational complexity, which means the training and testing phases are time-consuming. Secondly, the DSIC method requires the two stereo cameras to be on the same horizontal line. In the case when there is a shift on the vertical direction, the DSIC method may fail.

To overcome the aforementioned drawbacks of DSIC, we propose in this paper a homography matrix based SIC network, namely HESIC. As we know, the two stereo cameras

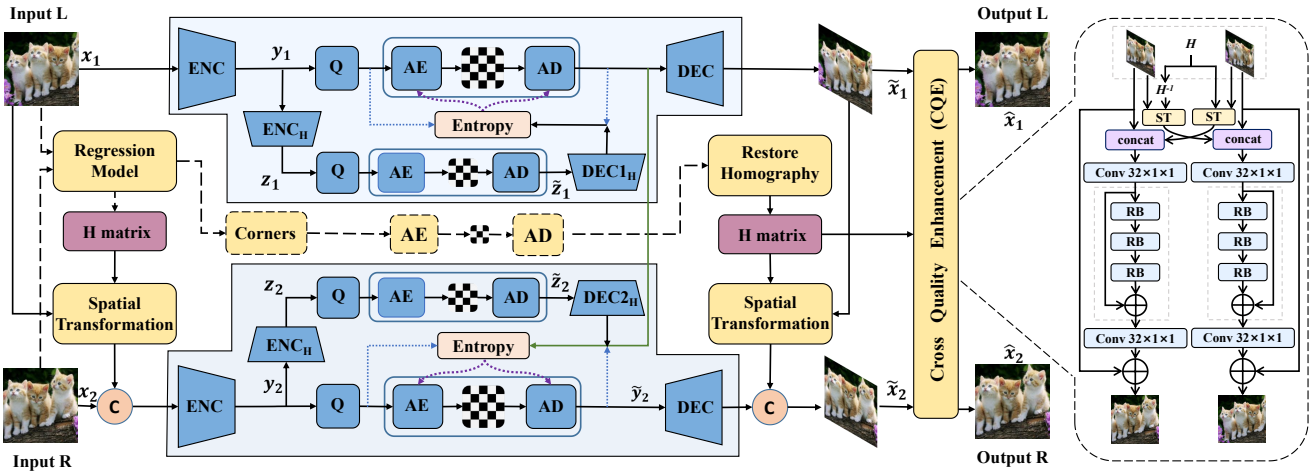


Figure 2. The overall network architecture of the proposed method. The left and right stereo images are jointly compressed to significantly save the bit-rates. Here, AE stands for arithmetic encoding, AD indicates arithmetic decoding, and ST is spatial transformation. Note that the blue dotted lines are only for HESIC+, while the others are for both HESIC and HESIC+.

usually take photos from two different angles, which makes homography transformation suitable for SIC. In addition, the calculation and encoding of homography matrix are quite computationally friendly, making it easy to be stored and transmit. Although homography matrix has been successfully used in tasks like image stitching [20] and light field compression [26], to the best of our knowledge, our work is the first attempt to appply it in SIC task. As demonstrated in experimental results, our HESIC provides much higher image quality than DSIC with less coding bit-rates.

3. Proposed method

3.1. Framework

Fig. 2 shows the overall framework of the proposed HESIC method. We first use a deep regression model to estimate the homography matrix (H matrix) between stereo images, and then the left image (denoted as $\boldsymbol{x}_1$) is spatially transformed by the H matrix to compensate the difference between $\boldsymbol{x}_1$ and the right image $\boldsymbol{x}_2$. The details of the regression model and H matrix will be introduced in Section 3.2.

After spatial transformation, $\boldsymbol{x}_1$ is compressed via an auto-encoder, and we concatenate $\boldsymbol{x}_2$ with the transformed $\boldsymbol{x}_1$ as the inputs to the second auto-encoder, which learns to compress the residual information between stereo images. Due to the correlation among $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, we model the probability function of the latent representation in the second auto-encoder (denoted as $\boldsymbol{y}_2$) conditioned on the latent representation of $\boldsymbol{x}_1$ (denoted as $\boldsymbol{y}_1$) to reduce the bit-rate. The information of the H matrix is also encoded into the bit-stream, and is used to transform the compressed left image ($\tilde{\boldsymbol{x}}_1$) at the decoder side. Then, the compressed residual output from the second auto-encoder is concatenated with the transformed $\tilde{\boldsymbol{x}}_1$ to reconstruct $\tilde{\boldsymbol{x}}_2$. The auto-encoder and probability model are to be detailed in Section 3.3. Finally, we propose a cross quality enhancement (CQE) network, which takes as inputs both the H matrix and compressed images to improve the compression quality of each image by using the correlated information in stereo images. The proposed CQE network is to be introduced in Section 3.4.

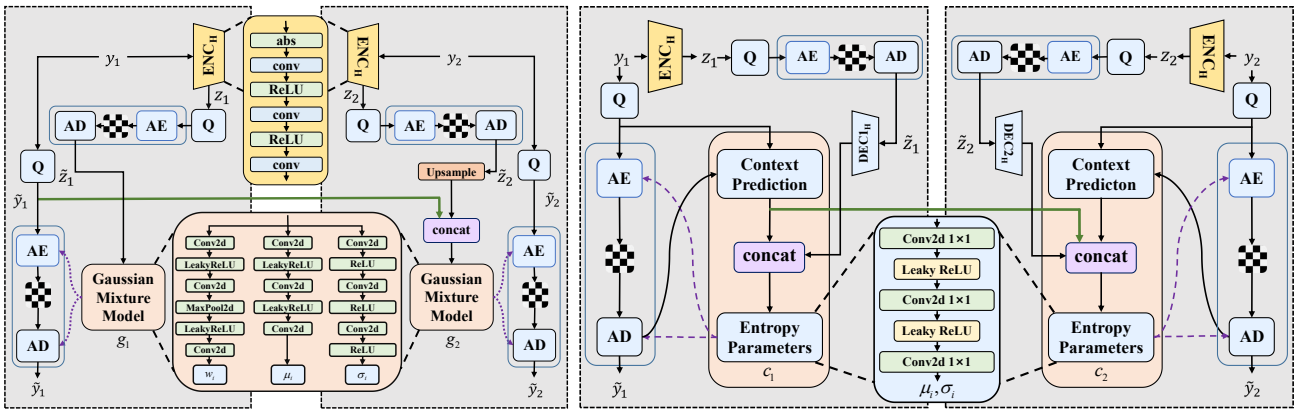
3.2. Homography matrix

To use the mutual information between stereo images, the first thing is to match the images or features. In DSIC [32], the two stereo features are matched by computing the weighted sum of feature vectors to obtain the similar content of images. Since the pixel-level dense warp is used to pass the mutual information, DSIC has high computational complexity. Actually, stereo images are usually taken at the same time from different angles, indicating that there is a spatial transformation relationship between them. Thus, we propose to use homography transformation to match stereo images in out method.

The homography transformation based image matching is flexible and lightweight, and the homography matrix (H matrix) is easy to be calculated and transmit with different conditions. Specifically, the coordinates of the point (u, v) in the left image can be transformed to the point (u', v') in the right image through the H matrix as follows,

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \mathbf{H} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}. \quad (1)$$

To achieve end-to-end learning, we adopt a deep-learning-based regression model [35] to generate the H matrix. The regression model is composed of several convolutional and fully connected layers. It firstly finds the corner coordinates of the two images and then calculate the H matrix. After we obtain the H matrix, a differentiable spatial transformation



(a) GMM-based stereo entropy model. (b) Context-based stereo entropy model.

Figure 3. Illustrations of the two conditional stereo entropy models in our HESIC and HESIC+ methods.

(ST) module [35, 25] is adopted to warp the left image to right image using the H matrix. Since the H matrix is a decimal matrix, the coordinates of the point (u', v') may not be integer if directly applying the H matrix in Eq. (1). Thus, in ST, the inverse H matrix is firstly calculated and then the coordinate (u', v') in the right image is reversely mapped to (u, v) in the left image. The pixel value in position (u', v') is obtained by averaging the surrounding pixels of (u, v) in the left image through bilinear interpolation as follows [35],

$$s^c = \sum_{i}^{M_I} \sum_{j}^{N_I} I_L^c(i,j) \max(0, 1-|u-i|) \max(0, 1-|v-j|), \tag{2}$$

where M_I,N_I are the height and width of the left image I_L, and $I_L^c(i,j)$ is the pixel value at location (i,j) in channel c of the left image. s^c is the warped pixel value at location (u',v') in channel c of the right image.

The transmit of H matrix is a problem in image compression since it may increase the coding bit-rates. Since corner coordinates and H matrix can convert to each other, we adopt to transmit the much lighter corner coordinates instead of the H matrix. We first convert the corner coordinates to integers and then encode them to store and transmit. For image with size 512×512, the bit-rate overhead for transmitting the corner coordinates is only $1.3 * 10^{-4}$ bits per pixel (bpp), which is negligible in image compression.

3.3. Auto-encoder and probability model

In the proposed HESIC method, we first use an auto-encoder to compress $\boldsymbol{x}_1$ as an independent image. Then, due to the high correlation between stereo images, we feed $\boldsymbol{x}_2$ concatenated with the spatially transformed $\boldsymbol{x}_1$ (by H matrix) to the second auto-encoder, as such it learns to compress the residual information between $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$. In the two auto-encoders, we utilize the same encoder, decoder and hyper transform networks as [4]. However, the single image compression method [4] models the probability mass function (PMF) of $\tilde{\boldsymbol{y}}$ only conditioned on the hyper-prior $\tilde{\boldsymbol{z}}$, *i.e.*, applying $q_{\tilde{\boldsymbol{y}}\,|\,\tilde{\boldsymbol{z}}}(\tilde{\boldsymbol{y}}\,|\,\tilde{\boldsymbol{z}})$[1] in entropy coding. In stereo image compression, we propose estimating $q_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2}(\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2)$ for the entropy coding of $\tilde{\boldsymbol{y}}_2$. Due to the high correlation between left and right images, conditioned on the information of $\tilde{\boldsymbol{y}}_1$, the (cross) entropy of $\tilde{\boldsymbol{y}}_2$ is expected to be smaller, thus resulting in lower bit-rate. In this paper, we utilize two conditional stereo entropy models, which are to be introduced in the following.

GMM-based entropy model. We first follow [32, 11] to estimate the conditional probability functions of $q_{\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1}$ and $q_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2}$ via Gaussian mixture models (GMMs) as follows,

$$\begin{aligned} q_{\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1} &\sim \sum_{n=1}^{N} \boldsymbol{w}_1^{(n)} \cdot \mathcal{N}(\boldsymbol{\mu}_1^{(n)}, \boldsymbol{\sigma}_1^{(n)}), \\ q_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2} &\sim \sum_{n=1}^{N} \boldsymbol{w}_2^{(n)} \cdot \mathcal{N}(\boldsymbol{\mu}_2^{(n)}, \boldsymbol{\sigma}_2^{(n)}), \end{aligned} \tag{3}$$

where N is the number of Gaussian functions. In (3), $\boldsymbol{w}$, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the parameters of the GMMs, in which $\{\boldsymbol{w}_1^{(n)}, \boldsymbol{\mu}_1^{(n)}, \boldsymbol{\sigma}_1^{(n)}\}_{n=1}^N$ and $\{\boldsymbol{w}_2^{(n)}, \boldsymbol{\mu}_2^{(n)}, \boldsymbol{\sigma}_2^{(n)}\}_{n=1}^N$ are generated by the deep networks g_1 and g_2, respectively. As shown in Fig. 3 (a), g_1 has the input of $\tilde{\boldsymbol{z}}_1$ and g_2 takes as inputs both $\tilde{\boldsymbol{y}}_1$ and $\tilde{\boldsymbol{z}}_2$. Hence, g_1 and g_2 learn to estimate $q_{\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1}$ and $q_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2}$ in (3), respectively. Given (3), the expected bit-rate for arithmetic coding [29] can be obtained as follows,

$$\begin{aligned} R = &\mathbb{E}_{\tilde{\boldsymbol{y}}_1 \sim p_{\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1}}[-\log_2 q_{\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1}(\tilde{\boldsymbol{y}}_1\,|\,\tilde{\boldsymbol{z}}_1)] \\ &+ \mathbb{E}_{\tilde{\boldsymbol{y}}_2 \sim p_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2}}[-\log_2 q_{\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2}(\tilde{\boldsymbol{y}}_2\,|\,\tilde{\boldsymbol{y}}_1,\tilde{\boldsymbol{z}}_2)], \end{aligned} \tag{4}$$

where p stands for true PMF.

Context-based entropy model. Inspired by [34] which successfully advances single image compression by adopting context-based entropy model, we introduce the auto-regressive context model into stereo image compression as

[1] Note that we define the true PMF as p and the estimated PMF as q.

another option for entropy coding. In addition to modelling the PMFs of $\tilde{\boldsymbol{y}}_1$ and $\tilde{\boldsymbol{y}}_2$ only conditioned on $\tilde{\boldsymbol{z}}_1$ and $\tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2$, respectively, the context-based model further describes the element-wise dependency within $\tilde{\boldsymbol{y}}_1$ and $\tilde{\boldsymbol{y}}_2$. That is, defining $\tilde{y}_1{}^i$ and $\tilde{y}_2{}^i$ as the i-th element in $\tilde{\boldsymbol{y}}_1$ and $\tilde{\boldsymbol{y}}_2$, respectively, $q_{\tilde{\boldsymbol{y}}_1 \mid \tilde{\boldsymbol{z}}_1}$ and $q_{\tilde{\boldsymbol{y}}_2 \mid \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2}$ can be formulated as

$$
\begin{aligned}
q_{\tilde{\boldsymbol{y}}_1 \mid \tilde{\boldsymbol{z}}_1}(\tilde{\boldsymbol{y}}_1 \mid \tilde{\boldsymbol{z}}_1) &= \prod_i q_{\tilde{y}_1{}^i \mid \tilde{y}_1^{<i}, \tilde{\boldsymbol{z}}_1}(\tilde{y}_1{}^i \mid \tilde{y}_1^{<i}, \tilde{\boldsymbol{z}}_1), \\
q_{\tilde{\boldsymbol{y}}_2 \mid \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2}(\tilde{\boldsymbol{y}}_2 \mid \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2) &= \prod_i q_{\tilde{y}_2{}^i \mid \tilde{y}_2^{<i}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2}(\tilde{y}_2{}^i \mid \tilde{y}_2^{<i}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2),
\end{aligned}
\tag{5}
$$

because of the chain rule of probability function.

To model the PMFs in (5), we feed $\tilde{\boldsymbol{y}}_1$ and $\tilde{\boldsymbol{y}}_2$ into the masked CNNs [41] which mask the elements $\tilde{y}^{\geq i}$ when calculating the features for the i-th element, and thus it is able to predict the PMFs conditioned on $\tilde{y}^{<i}$. Here we use Gaussian distributions to model the PMFs as follows

$$
\begin{aligned}
q_{\tilde{y}_1{}^i \mid \tilde{y}_1^{<i}, \tilde{\boldsymbol{z}}_1} &\sim \mathcal{N}(\mu_1^i, \sigma_1^i), \\
q_{\tilde{y}_2{}^i \mid \tilde{y}_2^{<i}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2} &\sim \mathcal{N}(\mu_2^i, \sigma_2^i).
\end{aligned}
\tag{6}
$$

As Fig. 3 (b) illustrates, the Gaussian parameters $\{\mu_1^i, \sigma_1^i\}_i$ and $\{\mu_2^i, \sigma_2^i\}_i$ are generated by the deep networks c_1 and c_2, which takes as inputs $\tilde{y}_1^{<i}, \tilde{\boldsymbol{z}}_1$ and $\tilde{y}_2^{<i}, \tilde{\boldsymbol{y}}_1, \tilde{\boldsymbol{z}}_2$, respectively. As such, the conditional PMFs in (5) can be estimated by our context-based stereo entropy model. Finally, given (5), the expected bit-rate can be calculated by (4).

3.4. Cross quality enhancement (CQE)

At the decoder side, we propose the CQE network to further improve the compression quality. Our CQE network can be trained jointly with the compression network, and thus can be seen as a component of our deep decoder. Fig 2 shows that the CQE network contains two sub-nets with the same structure to enhance each image, and each sub-net has 3 residual blocks with skip connections. Recall that the H matrix is learned by the regression model to describe the spatial difference from $\boldsymbol{x}_1$ to $\boldsymbol{x}_2$. Therefore, in the CQE network, we calculate the reverse H matrix (denoted as $\mathbf{H}^{-1}$ in Fig 2) to transform $\tilde{\boldsymbol{x}}_2$, and then feed $\tilde{\boldsymbol{x}}_1$ with the transformed $\tilde{\boldsymbol{x}}_2$ to the first sub-net to enhance $\tilde{\boldsymbol{x}}_1$. Similarly, the second sub-net takes as inputs both $\tilde{\boldsymbol{x}}_2$ and the $\tilde{\boldsymbol{x}}_1$ transformed by the original H matrix. As such, the proposed CQE network is able to utilize the correlation between stereo images for enhancing the compression quality of both images. We define the enhanced left and right images as $\hat{\boldsymbol{x}}_1$ and $\hat{\boldsymbol{x}}_2$, respectively.

3.5. Training strategy

In the training stage, we first pre-train the regression model to get the H matrix with the loss function as follows,

$$
\mathcal{L}_{\mathbf{H}} = D(\boldsymbol{x}_2\,,\, F_s(\boldsymbol{x}_1, F_r(\boldsymbol{x}_1, \boldsymbol{x}_2))), \tag{7}
$$

in which $F_r(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the regression model to calculate the H matrix, and $F_s(\boldsymbol{x}_1, H)$ indicates the spatial transformation by the H matrix. D denotes distortion which is defined as the Mean Square Error (MSE) in this paper. After pre-training, we design the loss function to train the whole network in an end-to-end manner. With R defined as the estimated joint coding bit-rate of the stereo images, the total loss function is defined as follows,

$$
\mathcal{L}_{\text{total}} = \lambda_d(D(\boldsymbol{x}_1\,,\, \hat{\boldsymbol{x}}_1) + D(\boldsymbol{x}_2\,,\, \hat{\boldsymbol{x}}_2)) + \lambda_r R, \tag{8}
$$

where λ_d and λ_r are the weights to the distortion and the bit-rate, respectively.

4. Experiments

4.1. Settings

Datasets. We evaluate the compression performance of our method on two public stereo image datasets: InStereo2K [5] and KITTI [21]. The InStereo2K dataset [5] consists of 2,050 pairs of stereo images, from which we randomly select 1,950 pairs for training and 50 pairs for validation. The remaining 50 pairs are used for testing. The KITTI dataset [21] provides the stereo images captured in the scenario of autonomous driving. To evaluate on KITTI, the models trained on InStereo2K are fine-tuned to the KITTI dataset. We randomly select 1,950,50, and 50 samples in KITTI for fine-tuning, validation, and testing respectively. The stereo images in InStereo2K [5] are with close views, while the images in KITTI [21] are with far-views. By using these two datasets, we can evaluate the compression performance of the proposed method more comprehensively.

Implementation. In this paper, we train two networks of the proposed HESIC method, which are with the GMM-based and context-based entropy models, respectively. They are defined as HESIC and HESIC+ in the following. The Adam optimizer [28] is adopted with standard parameters and learning rate of 10^{-4} in both the regression model training and the whole network training processes. The results shown in the experiments are obtained by training the network for 400 epochs. In the GMM-based model, we set $N = 5$ in (3). For various coding bit-rates, we fix λ_r to 1.0 in loss function (8), and smoothly adjust the weight of distortion λ_d from 0.001 to 0.1.

Evaluation. As mentioned in Section 3.5, the proposed HESIC and HESIC+ methods are optimized towards MSE. Hence, we evaluate the compression quality by peak signal-to-noise ratio (PSNR), but following most previous learned compression methods, we also report the results of the multi-scale structural similarity (MS-SSIM) [44] index. In addition, we compare the Bjøntegaard delta PSNR (BD-PSNR) [9] and BD-rate to assess the rate-distortion performance. Note that the higher value of BD-PSNR and lower

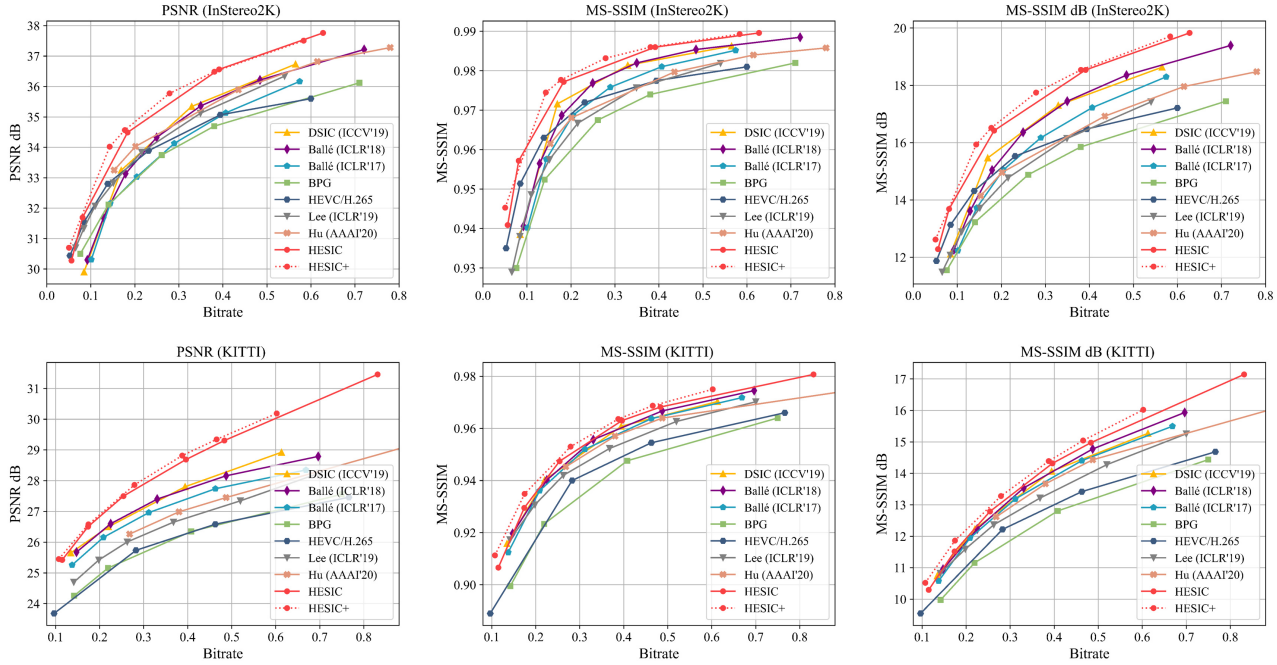


Figure 4. Rate–distortion curves for PSNR (dB), MS-SSIM and MS-SSIM (dB) with different compression methods.

value of BD-rate indicate better image compression performance. For the comparison methods, we compare our HESIC and HESIC+ models with both single image compression methods including Ballé *et al*. [3] (ICLR'17), Ballé *et al*. [4] (ICLR'18), Lee *et al*. [30] (ICLR'19), Hu *et al*. [24] (AAAI'20), and state-of-the-art stereo image compression method DSIC [32] (ICCV'19). Besides, we also compare our model with the traditional image and video codecs, such as BPG [7] and HEVC [39] (HM 16.20). Note that when comparing with HEVC, we feed stereo images as two video frames into the encoder.

4.2. Comparison against SOTA methods

Quantitative results. Table 1 presents the BD-PSNR and BD-rate results of our and the other comparison methods with Ballé (ICLR'18) [4] as the baseline. As we mentioned before, the higher value of BD-PSNR, and the lower value of BD-rate indicate better rate-distortion performance. As we can see from Table 1, our method achieves the highest BD-PSNR and lowest BD-rate on both InStereo2K and KITTI datasets, *i.e*., we use the smallest amount of bit-rates to achieve the best PSNR results. To intuitively show the compression performance of different methods, Fig. 4 plots the rate-distortion curves of our and the other comparison methods, in terms of PSNR, MS-SSIM and MS-SSIM (dB). Here, since the difference of MS-SSIM among different methods is not quite clear, we follow [32] to add MS-SSIM (dB) which is calculated by $10\log_{10}(1/(1-\text{MS-SSIM}))$. Please note that the bit-rate in this figure indicates the average bit-rate of the left and right images. As shown in this figure, the RD curve of our method is above the curves of other methods for both the two datasets. This demonstrates that our method outperforms other compression methods with the best rate-distortion performance. From Fig. 4, we can also note that although our model is only optimized towards PSNR, we still achieve the best compression performance among all methods in terms of MS-SSIM.

Table 1. BD-PSNR and BD-rate comparisons on different datasets, with the best results in red and second bests in blue.

InStereo2K dataset		
Methods	BD-PSNR (dB) ↑	BD-rate (%) ↓
Ballé (ICLR'17)	-0.489	14.195
BPG	-0.501	14.162
HEVC/H.265	-0.005	-11.342
Lee (ICLR'19)	0.192	-8.167
Hu (AAAI'20)	0.169	-4.415
DSIC (ICCV'19)	0.238	-7.062
HESIC	1.312	-32.946
HESIC+	1.373	-38.809
KITTI dataset		
Methods	BD-PSNR (dB) ↑	BD-rate (%) ↓
Ballé (ICLR'17)	-0.311	16.750
BPG	-1.418	105.068
HEVC/H.265	-1.367	105.804
Lee (ICLR'19)	-0.897	55.633
Hu (AAAI'20)	-0.677	41.406
DSIC (ICCV'19)	0.005	-4.027
HESIC	0.883	-25.967
HESIC+	0.920	-28.836

Qualitative results. To vividly show the compression performance of different methods, we visualize in Fig. 5

(a) Ground truth (b) Lee (ICLR'19) (c) Ballé (ICLR'18) (d) DSIC (ICCV'19) (e) HESIC

Figure 5. Visual comparisons of the compressed left and right images using our and the comparison methods including Lee (ICLR'19) [30], Ballé (ICLR'18) [4], and DISC (ICCV'19) [32]. Since the bpp value for a single image has no meaning for SIC task, we present here the average bpp of the left and right images.

Table 2. Computational complexity of our and DSIC method

Method	Network FLOPs	Params	Enc-time	Dec-time
DSIC	766.4G	91.5M	322.06 ms	261.57 ms
HESIC	212.5G	69.3M	176.49 ms	174.38 ms
HESIC+	191.1G	50.6M	186.80 ms	8878.89 ms

the compressed left and right images using our and other compression methods. To make the comparison fair, all images are compressed with similar bit-rate per pixel (bpp). As can be seen from this figure, our method is able to use less bit-rates to achieve higher PSNR values for both left and right images. For the image quality, our method is capable to clearly restore the details of the stripes on the tiger and the lines on the basketball, while the comparison methods lead to either blurred details [4] or ringing around edges [30].

Computational complexity. Table 2 compares the FLOPs, number of network parameters, encoding and decoding time of DSIC and our two models. For fair comparison, the encoding and decoding time is tested on a RTX2070s GPU for both DSIC and our method. As we can see from Table 2, the FLOPs of both our HESIC and HESIC+ are around 3 times smaller than DSIC, while the FLOPS of HESIC+ is further smaller than HESIC. The similar phenomenon can be seen in network parameters. For encoding and decoding time, our HESIC is more than 2 times faster than DSIC. This

shows that our method with GMM entropy model is more computational friendly, which is suitable for practical applications. However, our HESIC+ has larger decoding time due to the auto-regressive entropy model, which calculates the PMF of $\tilde{y}^i$ from its previous elements $\tilde{y}^{<i}$, and therefore it fails to be speed up by GPU parallel computation. The high decoding time is the expense of better compresssion performance, i.e., our HESIC+ achieves the best compression performance as shown in Table 1 and Fig. 4. In other words, we provide here two options for stereo image compression: the HESIC model is more suitable if decoding speed is an important consideration, while the HESIC+ model is more appropriate if image quality is the most critical thing.

4.3. Ablation study

In this subsection, we implement several experiments to investigate the effect of mutual entropy connection, H matrix and the cross quality enhancement on the compression performance of our method.

Case 1: Effectiveness of mutual entropy connection. As shown in Fig. 2, there is a green line connecting $\tilde{\boldsymbol{y}}_1$ from the first entropy model to the second one. We call this line mutual entropy connection here. In order to verify its effectiveness, we simply remove it and then the model is equivalent to using two independent auto-encoders but with a residual image as input. Then, we retrain the model without the green line and denote this model as Case 1. As shown in Fig. 6, the RD curves of Case 1 are significantly lower than our original model, indicating the effectiveness of the mutual entropy connection. With the mutual entropy connection, the information of left image can help the right image, resulting in better compression performance.

Case 2: Effectiveness of homography matrix. To verify the effectiveness of homography matrix, we remove the regression model and spatial transformation in Fig. 2. In other words, the stereo images are input to the encoder directly without the residual image. After retraining the model, we can have its RD-curves in Fig. 6. As we can see, the RD curves of Case 2 is almost the lowest among all curves. This result demonstrates the H matrix plays a critical role in improving the stereo image coding performance.

Case 3: Effectiveness of CQE. The CQE is a part of the decoder, and we remove it to verify its effectiveness. As shown in Fig. 6, the RD-curves of Case 3 are lower than our original model with CQE. This demonstrates that the CQE indeed helps improve our coding performance. In addition, note that even without CQE, our method still performs better than the comparison methods, *i.e.*, the RD curves of Case 3 are higher than that of Ballé (ICLR'18). Considering the position of Ballé (ICLR'18) in Fig. 4, we can conclude that our method without CQE still outperforms other methods.

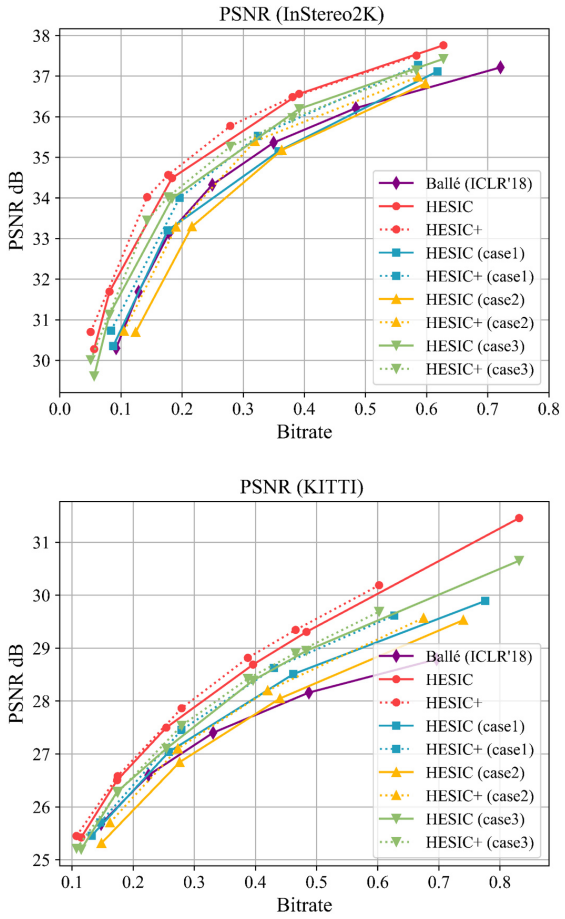


Figure 6. Rate-distortion curves for ablation study.

5. Conclusion

In this paper, we propose a novel homography transform based deep neural network for stereo image compression, which drastically increases stereo image quality with less coding bit-rates. Firstly, a light homography matrix (H matrix) is calculated through a regression model, which maps the left image to right image to get the residual image. Then, we use two conditional entropy models, *i.e.*, Gaussian mixture model based entropy model and context-based entropy model, to jointly encode the two stereo images. Finally, at the decoder, we propose a cross quality enhancement module to further enhance the compressed image quality through H and inverse H matrices. Experimental results verify the effectiveness of our method through exhaustive comparisons and ablation studies.

Acknowledgments

This work was sponsored by CAAI-Huawei Mindspore Open Fund, NSFC under Grants 62050175, 62001016, 61876013, and 61922009, and Beijing Natural Science Foundation under Grant JQ20020.

References

[1] Alekh Karkada Ashok and Nagaraju Palani. Autoencoders with Variable Sized Latent Vector for Image Compression. In *CVPR Workshops*, pages 2547–2550, 2018. 2

[2] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. Density modeling of images using a generalized normalization transformation. *arXiv preprint arXiv:1511.06281*, 2015. 2

[3] Johannes Ballé, Valero Laparra, and Eero P Simoncelli. End-to-end optimized image compression. *arXiv preprint arXiv:1611.01704*, 2016. 2, 6

[4] 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, 2, 4, 6, 7

[5] Wei Bao, Wei Wang, Yuhua Xu, Yulan Guo, Siyu Hong, and Xiaohu Zhang. Instereo2k: a large real dataset for stereo matching in indoor scenes. *Science China Information Sciences*, 63(11):1–11, 2020. 5

[6] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *European Conference on Computer Vision*, pages 404–417. Springer, 2006. 2

[7] Fabrice Bellard. BPG image format. *URL https://bellard.org/bpg*, 1, 2015. 2, 6

[8] I Bezzine, Mounir Kaaniche, Saadi Boudjit, and Azeddine Beghdadi. Sparse optimization of non separable vector lifting scheme for stereo image coding. *Journal of Visual Communication and Image Representation*, 57:283–293, 2018. 1

[9] Gisle Bjontegaard. Calculation of average PSNR differences between RD-curves. *VCEG-M33*, 2001. 5

[10] Matthew Brown, David G Lowe, et al. Recognising panoramas. In *Proceedings of the IEEE International Conference on Computer Vision*, volume 3, page 1218, 2003. 2

[11] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and Jiro Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7939–7948, 2020. 4

[12] Daniel B Chuang, Lawrence M Candell, William D Ross, Mark E Beattie, Cindy Y Fang, Bobby Ren, and Jonathan P Blanchard. Imaging system for immersive surveillance, May 19 2015. US Patent 9,036,001. 1

[13] Ondřej Chum, Jiří Matas, and Josef Kittler. Locally optimized ransac. In *Joint Pattern Recognition Symposium*, pages 236–243. Springer, 2003. 2

[14] Xin Deng and Pier Luigi Dragotti. Deep coupled ista network for multi-modal image super-resolution. *IEEE Transactions on Image Processing*, 29:1683–1698, 2019. 2

[15] Xin Deng and Pier Luigi Dragotti. Deep convolutional neural network for multi-modal image restoration and fusion. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 2

[16] Xin Deng, Yutong Zhang, Mai Xu, Shuhang Gu, and Yiping Duan. Deep coupled feedback network for joint exposure fusion and image super-resolution. *IEEE Transactions on Image Processing*, 30:3098–3112, 2021. 2

[17] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Deep image homography estimation. *arXiv preprint arXiv:1606.03798*, 2016. 2

[18] JN Ellinas and Manolis S Sangriotis. Stereo image compression using wavelet coefficients morphology. *Image and Vision Computing*, 22(4):281–290, 2004. 1

[19] Christoph Fehn. Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV. In *Stereoscopic Displays and Virtual Reality Systems XI*, volume 5291, pages 93–104. International Society for Optics and Photonics, 2004. 1

[20] Junhong Gao, Seon Joo Kim, and Michael S Brown. Constructing image panoramas using dual-homography warping. In *CVPR 2011*, pages 49–56. IEEE, 2011. 3

[21] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research*, 32(11):1231–1237, 2013. 5

[22] Vivek K Goyal. Theoretical foundations of transform coding. *IEEE Signal Processing Magazine*, 18(5):9–21, 2001. 2

[23] Jianhua Hu, Ming Li, Changsheng Xia, and Yundong Zhang. Combine Traditional Compression Method With Convolutional Neural Networks. In *CVPR Workshops*, pages 2563–2566, 2018. 2

[24] Yueyu Hu, Wenhan Yang, and Jiaying Liu. Coarse-to-fine hyper-prior modeling for learned image compression. In *AAAI*, pages 11013–11020, 2020. 6

[25] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *Advances in Neural Information Processing Systems*, pages 2017–2025, 2015. 2, 4

[26] Xiaoran Jiang, Mikaël Le Pendu, Reuben A Farrugia, and Christine Guillemot. Light field compression with homography-based low-rank approximation. *IEEE Journal of Selected Topics in Signal Processing*, 11(7):1132–1145, 2017. 3

[27] Aysha Kadaikar, Gabriel Dauphin, and Anissa Mokraoui. Joint disparity and variable size-block optimization algorithm for stereoscopic image compression. *Signal Processing: Image Communication*, 61:1–8, 2018. 1

[28] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2015. 5

[29] Glen G Langdon. An introduction to arithmetic coding. *IBM Journal of Research and Development*, 28(2):135–149, 1984. 4

[30] 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. 6, 7

[31] 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. 2

[32] Jerry Liu, Shenlong Wang, and Raquel Urtasun. DSIC: Deep stereo image compression. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3136–3145, 2019. 1, 2, 3, 4, 6, 7

[33] David G Lowe. Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision*, volume 2, pages 1150–1157. IEEE, 1999. 2

[34] 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. 2, 4

[35] Ty Nguyen, Steven W Chen, Shreyas S Shivakumar, Camillo Jose Taylor, and Vijay Kumar. Unsupervised deep homography: A fast and robust homography estimation model. *IEEE Robotics and Automation Letters*, 3(3):2346–2353, 2018. 1, 2, 3, 4

[36] Siyuan Shan, Wen Yan, Xiaoqing Guo, Eric I Chang, Yubo Fan, Yan Xu, et al. Unsupervised end-to-end learning for deformable medical image registration. *arXiv preprint arXiv:1711.08608*, 2017. 2

[37] Athanassios Skodras, Charilaos Christopoulos, and Touradj Ebrahimi. The jpeg 2000 still image compression standard. *IEEE Signal Processing Magazine*, 18(5):36–58, 2001. 2

[38] Hirschmuller H Stereo. Processing by Semiglobal Matching and Mutual Information [J]. *IEEE Transactions on Pattern Analysis & Machine Intelligence*, 30(2):328–341, 2007. 2

[39] Gary J Sullivan, Jens-Rainer Ohm, Woo-Jin Han, and Thomas Wiegand. Overview of the high efficiency video coding (HEVC) standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12):1649–1668, 2012. 2, 6

[40] 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. 2

[41] Aaron Van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, et al. Conditional image generation with pixelcnn decoders. In *Advances in Neural Information Processing Systems*, pages 4790–4798, 2016. 5

[42] Gregory K Wallace. The JPEG still picture compression standard. *IEEE Transactions on Consumer Electronics*, 38(1):xviii–xxxiv, 1992. 2

[43] Yaolong Wang, Mingqing Xiao, Chang Liu, Shuxin Zheng, and Tie-Yan Liu. Modeling Lost Information in Lossy Image Compression. *arXiv preprint arXiv:2006.11999*, 2020. 2

[44] Zhou Wang, Eero P Simoncelli, and Alan 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. 5

[45] Huan Yin, Yue Wang, Li Tang, Xiaqing Ding, Shoudong Huang, and Rong Xiong. 3D LiDAR Map Compression for Efficient Localization on Resource Constrained Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 2020. 1

[46] Jure Žbontar and Yann LeCun. Stereo matching by training a convolutional neural network to compare image patches. *The Journal of Machine Learning Research*, 17(1):2287–2318, 2016. 2

[47] Lei Zhou, Chunlei Cai, Yue Gao, Sanbao Su, and Junmin Wu. Variational Autoencoder for Low Bit-rate Image Compression. In *CVPR Workshops*, pages 2617–2620, 2018. 2