MoCha-Stereo: Motif Channel Attention Network for Stereo Matching

Supplementary Material

1. Implementation Details of MoCha-MVS

MoCha-MVS was trained and tested using a configuration similar to that of most MVS methods [3, 10, 13] on the DTU dataset[4]. Similar to other learning-based MVS [3, 10, 13], the output depth map is filtered based on photometric and geometric consistencies. To save computational costs, MoCha-MVS utilizes only 3 iterations, and is trained on the DTU training set [4] for 2 days with a batch size of 4 on single NVIDIA Tesla A6000 GPU.

2. More Details on the Structure of MoCha-MVS & MoCha-Stereo

2.1. Loss Function of MoCha-MVS

During the training phase, our approach produces multiple depth maps from both the initial depth prediction module and the multi-stage LSTM-based optimization module, each corresponding to different iteration steps. Subsequently, we compute L1 losses between all output depth maps and their respective ground truth depth maps of matching resolution. The final loss is then computed as a weighted sum of all these losses as Equ. 1.

$$L = L_0 + \sum_{k=0}^{2} \sum_{i=1}^{T_k+1} \lambda_i^k L_i^k \tag{1}$$

where L_0 represents the loss associated with the initial depth map generated by the MVS-version MCCV. T_k denotes the number of optimization iterations at stage k. $\{L_i^k | i = 1 \ldots T_k + 1\}$ refers to the losses of T_k output depth maps and an upsampled depth map at stage k, and λ_i^k represents the corresponding weight.

2.2. Feature Extractor

MoCha-Stereo. Following [12] and MoCha-MVS, we also utilizes a frozen network as feature encoder. Feature Extractor of MoCha-Stereo extract the multi-scale features $f_{2^i}^{encoder} \in \mathbb{R}^{i=2,3,4,5}$ at $1/4$, $1/8$, $1/16$, and $1/32$ scales of the original shape first. It is worth noting that stereo matching is a task that relies on matching based on epipolar constraints, requiring the network to have strong local discriminative abilities. Despite the pre-trained model exhibits strong semantic understanding capabilities, the external features [2] provides exhibit limitations in capturing local information relevant to the stereo matching task. To bolster the local discriminative capability of the feature extractor, we also use the learnable decoder as shown in Equ. 2.

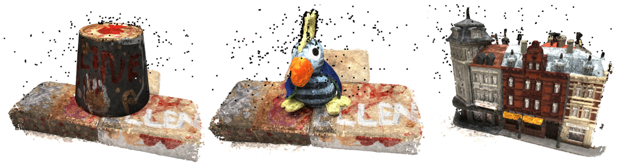

Figure 1. Visualization results of MoCha-MVS on a few DTU scenes [4].

$$f_{2^i}^{decoder} = Conv(Concat(f_{2^{i+1}}^{encoder}, Deconv(f_{2^i}^{decoder}))) \tag{2}$$

where $Deconv$ denotes the deconvolution operation, it deconvolution $f_{2^i}^{decoder}$ to match the resolution of $f_{2^{i+1}}^{encoder}$. This module implements a coarse-to-fine strategy by concatenating the features garnered from the deconvolution of the feature map at a lower resolution with the feature map at the same resolution, thus generating a new feature map at the current resolution level. This enhancement equips the feature extractor with the ability to learn internal features based on prior knowledge from external features, thereby incorporating multi-scale information to attain a more accurate semantic context.

MoCha-MVS. MoCha-Stereo is expanded to include MVS, or MoCha-MVS. The overall design of MoCha-MVS draws inspiration from CasMVSNet [3], IterMVS [10], and EffiMVS [11]. Following MoCha-Stereo, MoCha-MVS also use a frozen encoder to extract multi-scale features. We utilizes EfficientNet [8] pretrained on ImageNet [1] as feature encoder, and extract the multi-scale features $f_{2^i}^{encoder} \in \mathbb{R}^{i=2,3}$ at $1/4$, $1/8$, and $1/16$ scales of the original shape. However, the decoder of MoCha-MVS consists of only a single-layer convolution applied to features at different scales, which is different from MoCha-Stereo.

2.3. Sliding Window for Motif Channel Mining

The Sliding Window SW is a critical component in the acquisition of the Motif Channel. Through the interaction between the Sliding Window SW and the high frequency feature f_{fre}, we obtain repeated channel features f_{fre}^{mc}. We have found that the most appropriate implementation of the sliding window is a 3×3 convolution. This approach is actually consistent with our Equ. 3 in the main text.

$$f_{fre}^{mc}(s,h,w) = \sum_{c=1}^{N_c}\sum_{i=1}^{3}\sum_{j=1}^{3} (SW(s,h+i,w+j) \times f_{fre}(c,h,w)) \tag{3}$$

Because convolution in the spatial domain is equivalent to multiplication in the frequency domain, i.e., Equ. 4 holds true.

$$Conv_{3\times 3}(h,w) \triangleq \sum_{i=1}^{3}\sum_{j=1}^{3}(SW(h+i,w+j) \times f_{fre}(h,w)) \tag{4}$$

The Equ. 4 can be proven as follows:
Let x and y be spatial domain sequences, where both x and y are column vectors. Let X and Y be frequency domain sequences, i.e., the column vector features obtained by applying the Fourier transform F to x and y. Equ. 5 and Equ. 6 hold true here.

$$X = Fx \tag{5}$$

$$x = F^{-1}X \tag{6}$$

where the F can be represented by Equ. 7.

$$F = \begin{bmatrix} 1 & 1 & \dots & 1 \\ 1 & e^{-\frac{i2\pi}{N}} & \dots & e^{-\frac{(N-1)\cdot i2\pi}{N}} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & e^{-\frac{(N-1)\cdot i2\pi}{N}} & \dots & e^{-\frac{(N-1)^2\cdot i2\pi}{N}} \end{bmatrix} \tag{7}$$

To prove Equ. 4, it suffices to prove Equ. 8.

$$F \cdot Toep(x) \cdot y = Diag(X) \cdot F \cdot y \tag{8}$$

$$F \cdot Toep(x) \cdot F^{-1} = Diag(X) \tag{9}$$

where $Diag(X)$ denote the diagonal matrix with the elements of vector X as its diagonal elements, and $Toep(x)$ represent the Toeplitz matrix formed by arranging the elements of vector x. At this point, all y's make Equ. 8 hold. So to prove that Equ. 4 holds, all you need to do is prove that Equ. 9 holds.

The characteristics of each column of F^{-1} are such that each element is equal to the previous element multiplied by $e^{\frac{k\cdot i2\pi}{N}}$ (where k denotes the column index, starting from 0). The first element 1 is also equal to the last element $e^{\frac{k(N-1)\cdot i2\pi}{N}}$ multiplied by $e^{\frac{k\cdot i2\pi}{N}}$ (with the exponential function having a period of $i2\pi$). Due to the shifting relationship between rows in $Toep(x)$, when the $j+1$ row of $Toep(x)$ is multiplied by the k column of F^{-1}, it is equal to the j row of $Toep(x)$ multiplied by the k column of F^{-1} and then multiplied by $e^{\frac{k\cdot i2\pi}{N}}$. This proves that each column of F^{-1} is a characteristic vector of $Toep(x)$. Since the first element of each column of F^{-1} is 1, the characteristic value corresponding to the k column is the first row of $Toep(x)$ multiplied by the k column of F^{-1}, resulting in Equ. 10.

$$\lambda = x_0 + x_{n-1}e^{\frac{k\cdot i2\pi}{N}} + \cdots + x_{n-1}e^{\frac{-k\cdot(N-1)\cdot i2\pi}{N}} \tag{10}$$

$$\lambda = x_0 + x_1 e^{\frac{-k\cdot i2\pi}{N}} + \cdots + x_{n-1}e^{\frac{-k\cdot(N-1)\cdot i2\pi}{N}} \tag{11}$$

Due to the periodicity of the exponential function, Equ. 11 is satisfied. This means that X is equal to the k-th element of Fx. Therefore, the characteristic vectors of $Toep(x)$ are the columns of F^{-1}, and the characteristic values are the elements of X. Based on this conclusion, it can be inferred that Equ. 9 holds, which implies that Equ. 4 holds.

2.4. Iterative Update Operator

Following [5, 12, 14], we employ a Iterative Update Operator to update the initial disparity map obtained from the Motif Channel Correlation Volume (MCCV) in a coarse-to-fine manner. This kind of incremental learning is beneficial in mitigating catastrophic forgetting during the training process. We reproduced the iterative structures of ConvLSTM to ensure stable updates of features.

For each iteration, MoCha-Stereo or MoCha-MVS uses the current disparity d_t to index from the MCCV via linear interpolation. Then we use ConvLSTM to update the hidden state h_{t-1} and C_{t-1} as Equ. 12.

$$\begin{aligned} f_t &= \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \\ i_t &= \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ C_t' &= tanh(W_i \cdot [h_{t-1}, x_t] + b_c) \\ C_t &= f_t \times C_{t-1} + i_t \times C_t' \\ o_t &= \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t &= o_t \times tanh(C_t) \end{aligned} \tag{12}$$

where W_f, W_i represent the weights adaptively learned by the convolutional neural network.

2.5. Acquiring inputs for Unet of Reconstruction Error Motif Penalty Module

The final disparity map output from Iterative Update Operator is produced by upsampling [9] the disparity map at $1/4$ resolution, which results in some loss of spatial information. The loss of spatial structure can be compensated for through the reconstruction error. The reconstruction error is a crucial metric in various domains, capturing the disparity between true values and model predictions. Therefore, we incorporate the additional reconstruction error as input to the Refinement module to refine the full-resolution disparity map. The formula for the calculation of the reconstruction error in MoCha-Stereo is denoted as Equ. 7 in the original

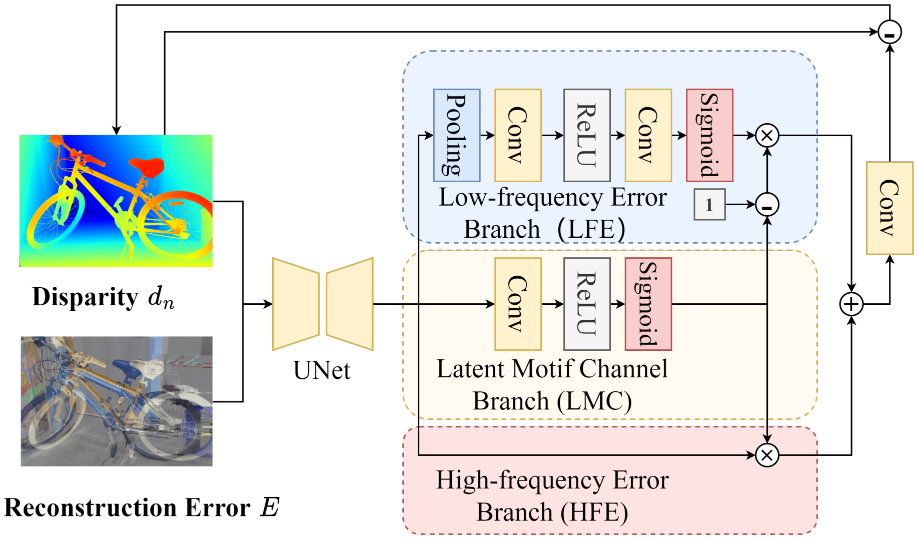
Disparity d_n
Reconstruction Error E
UNet
Pooling
Conv
ReLU
Conv
Sigmoid
1
Low-frequency Error Branch (LFE)
Conv
ReLU
Sigmoid
Latent Motif Channel Branch (LMC)
High-frequency Error Branch (HFE)
Conv

Figure 2. REMP module for Full-Resolution Refine. The upper branch above obtains low-frequency information through pooling, the lower branch retains the original high-resolution image as high-frequency detailed information, and the middle branch learns motif features through CNN.

manuscript, and here we represent it as Equ. 13.

$$E = K_l(R - \frac{TN^T}{D})K_r^{-1}I_r - I_l \tag{13}$$

We utilizes the homography transformation described in Equ. 14 to warp the right view into the coordinate system of the left view.

$$I_r^{warp} = K_l(R - \frac{TN^T}{D})K_r^{-1}I_r \tag{14}$$

$K_{l(r)}$ represents the intrinsic matrix of the left (right) camera in the stereo system, R is the rotation matrix from the right view coordinate system to the left view coordinate system, T is the translation matrix from the right view coordinate system to the left view coordinate system, N is the normal vector of the object plane in the right view coordinate system, D is the perpendicular distance between the object plane and the camera light source (this distance is obtained from the computed disparity).

$$EM = I_r^{warp} - I_l \tag{15}$$

$$input = REMP(Concat(Conv(IM), d_n)) \tag{16}$$

EM is the error map, d_n represents the initial disparity map before undergoing the homography processing, I_r is the right view image, and $REMP$ means our Reconstruction Error Motif Penalty (REMP) module, as shown in Fig. 2. For MoCha-MVS, the reconstruction error can be expressed using Formula 17.

$$E_{mvs} = \sum_{i=1}^{v}(K_{ref}(R - \frac{TN^T}{D})K_i^{-1}I_i - I_{ref}) \tag{17}$$

where v means numbers of source images, I_{ref} means reference image.

3. Visualizations for MoCha-Stereo

3.1. Visualizations before and after optimizing channel features using Motif Channel Attention (MCA) for feature maps of stereo matching

One set of relationships affected by MCA is illustrated in Fig. 3, as already presented in the original manuscript. Additional correspondences are depicted in the following figures.

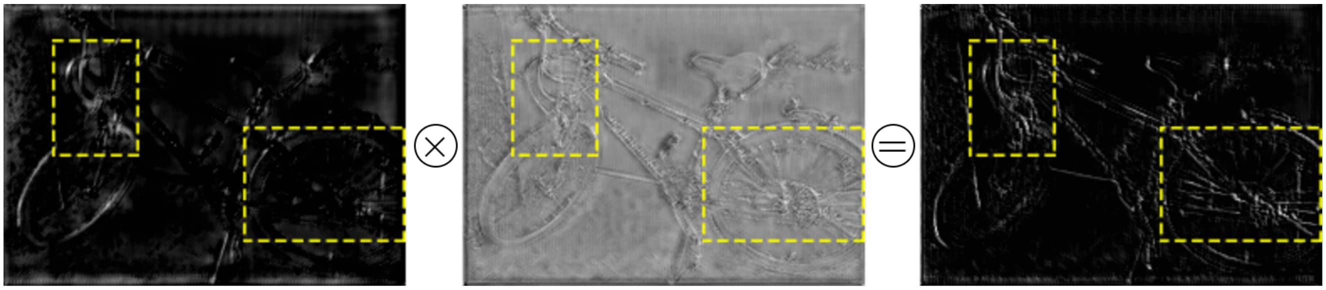

Figure 3. An example of one of the feature channels in visual form. The first picture shows the initial normal channel, and the last picture shows the visualization after paying attention to Motif Channels. The middle picture visualize a motif channel. It can be observed that the edge texture details are emphasized in the new feature channels.

3.2. Visualizations of Zero-shot performance

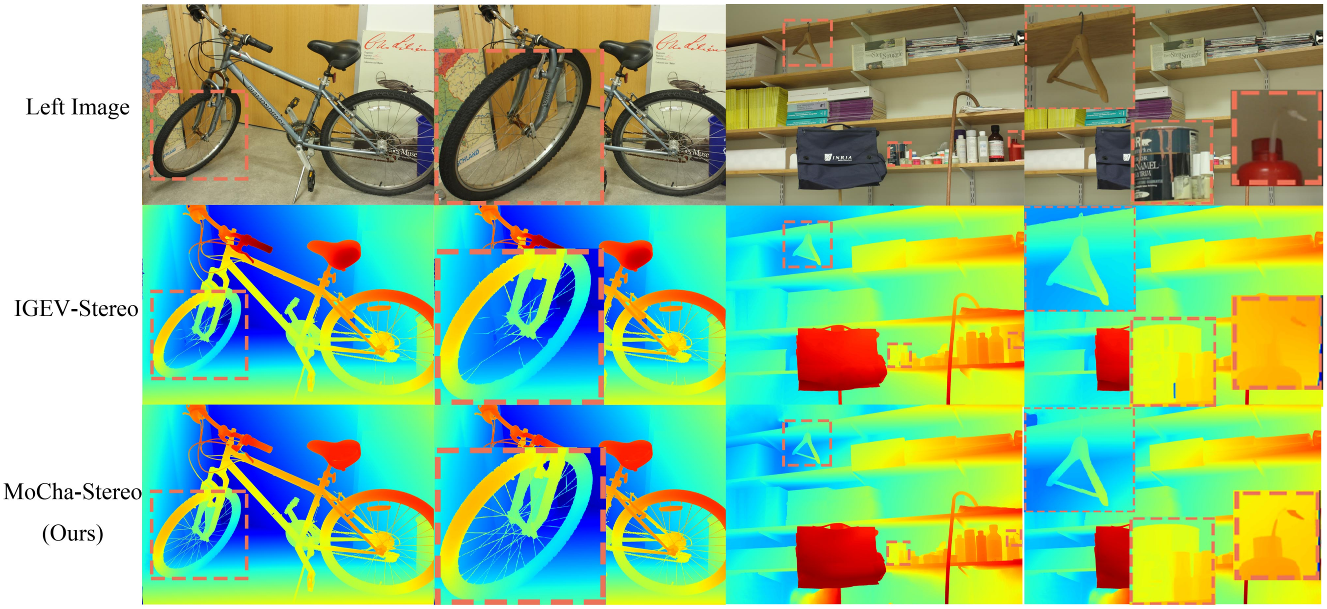

Figure 4. Comparisons on the Middlebury dataset [7]. All results presented in this section demonstrate zero-shot generalization on the Scene Flow dataset [6]. The odd-numbered columns show the original images, while the even-numbered columns present zoomed-in details for better visualization.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[2] Yikang Ding, Qingtian Zhu, Xiangyue Liu, Wentao Yuan, Haotian Zhang, and Chi Zhang. Kd-mvs: Knowledge distillation based self-supervised learning for multi-view stereo. In *European Conference on Computer Vision*, pages 630–646. Springer, 2022. 1

[3] Xiaodong Gu, Zhiwen Fan, Siyu Zhu, Zuozhuo Dai, Feitong Tan, and Ping Tan. Cascade cost volume for high-resolution multi-view stereo and stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2020. 1

[4] Rasmus Jensen, Anders Dahl, George Vogiatzis, Engin Tola, and Henrik Aanæs. Large scale multi-view stereopsis evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 406–413, 2014. 1

[5] Lahav Lipson, Zachary Teed, and Jia Deng. Raft-stereo: Multilevel recurrent field transforms for stereo matching. In *2021 International Conference on 3D Vision (3DV)*, pages 218–227. IEEE, 2021. 2

[6] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4040–4048, 2016. 4

[7] Daniel Scharstein, Heiko Hirschmüller, York Kitajima, Greg Krathwohl, Nera Nešić, Xi Wang, and Porter Westling. High-resolution stereo datasets with subpixel-accurate ground truth. In *Pattern Recognition: 36th German Conference, GCPR 2014, Münster, Germany, September 2-5, 2014, Proceedings 36*, pages 31–42. Springer, 2014. 4

[8] Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019. 1

[9] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 402–419. Springer, 2020. 2

[10] Fangjinhua Wang, Silvano Galliani, Christoph Vogel, and Marc Pollefeys. Itermvs: iterative probability estimation for efficient multi-view stereo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8606–8615, 2022. 1

[11] Shaoqian Wang, Bo Li, and Yuchao Dai. Efficient multi-view stereo by iterative dynamic cost volume. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8655–8664, 2022. 1

[12] Gangwei Xu, Xianqi Wang, Xiaohuan Ding, and Xin Yang. Iterative geometry encoding volume for stereo matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21919–21928, 2023. 1, 2

[13] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European conference on computer vision (ECCV)*, pages 767–783, 2018. 1

[14] Haoliang Zhao, Huizhou Zhou, Yongjun Zhang, Yong Zhao, Yitong Yang, and Ting Ouyang. Eai-stereo: Error aware iterative network for stereo matching. In *Proceedings of the Asian Conference on Computer Vision*, pages 315–332, 2022. 2