BALF: Simple and Efficient Blur Aware Local Feature Detector

Zhenjun Zhao
The Chinese University of Hong Kong
ericzzj89@gmail.com

Abstract

Local feature detection is a key ingredient of many image processing and computer vision applications, such as visual odometry and localization. Most existing algorithms focus on feature detection from a sharp image. They would thus have degraded performance once the image is blurred, which could happen easily under low-lighting conditions. To address this issue, we propose a simple yet both efficient and effective keypoint detection method that is able to accurately localize the salient keypoints in a blurred image. Our method takes advantages of a novel multi-layer perceptron (MLP) based architecture that significantly improve the detection repeatability for a blurred image. The network is also light-weight and able to run in real-time, which enables its deployment for time-constrained applications. Extensive experimental results demonstrate that our detector is able to improve the detection repeatability with blurred images, while keeping comparable performance as existing state-of-the-art detectors for sharp images. The code and trained weights are publicly available at github.com/ericzzj1989/BALF.

1. Introduction

Being able to accurately detect and describe salient keypoints across images is crucial in many applications such as Simultaneous Localization and Mapping (SLAM), Structure-from-Motion (SfM), camera calibration, video compression, tracking, image retrieval, and visual localization. Keypoints should be sparse, repeatable, and discriminable, in order to be extracted and matched across images from different lighting conditions or viewpoints. While many state-of-the-art methods have been proposed, motion blur is still a major challenge remaining for local feature detection methods. Motion blur is one of the most common artifacts that degrade images. It usually occurs in low-light conditions where longer exposure times are necessary. This would affect many feature based approaches, which struggle to detect repeatable keypoints to build up correspondences.

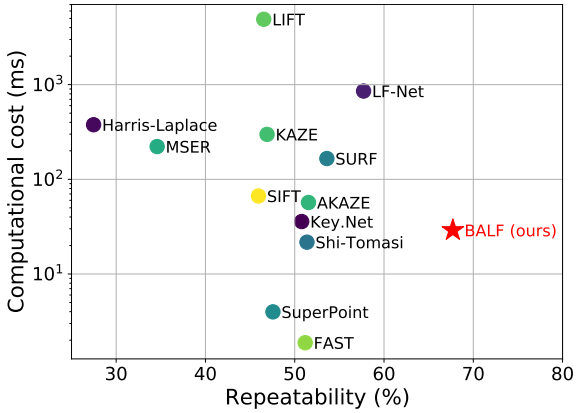

Figure 1. **The performance of keypoint detectors with motion blurred images.** Our approach achieve superior detection performance in terms of the repeatability metric and efficiency, which paves the way for robust 3D vision under low-lighting conditions.

To detect keypoints in blurred images, a straightforward way is to utilize a deblurring algorithm to restore the latent sharp image and then to detect keypoints from the restored image. Image/video deblurring methods have been well developed over the last decades, which mainly consist of classic gradient-descent methods [4, 9, 16, 22, 42, 43, 54, 55, 69, 71] and learning-based methods [20, 23, 24, 28, 33, 40, 53, 57, 60, 67, 70, 72, 77]. Although deblurring algorithms have achieved impressive performance recently, there are still several limitations existed. For example, existing state-of-the-art methods usually require high computational resources and are hardly to run in real-time even with a high-end GPU. Another limitation is that current methods still cannot perform very well and might introduce additional artifacts for severe motion blurred image, due to the limited information preserved by a single blurred image. We thus aim to design a novel one-stage efficient local feature detector from a motion blurred image directly, without any intermediate deblurring operation, to avoid those drawbacks.

In recent years, deep learning techniques have shown great success in improving local feature extraction. Many state-of-the-art methods have been proposed by the community [7, 25, 51, 58, 61, 65, 79]. Even though great progress

has been achieved in extracting local features, most of them focus on improving the robustness against viewpoint changes, illumination changes etc. Prior work that aims to extract features from motion blurred images is still limited, which is important towards robust 3D vision at low-lighting scenarios (*e.g*. augmented reality for outdoors at night). Further inspired by the recent success of Multi-Layer-Perceptron (MLP) in many areas of computer vision [8, 10, 18, 19, 29, 30, 32, 35, 59, 62–64, 75], we propose to explore the possibility by applying MLP-based network for local feature detection from a motion blurred image, which has never been attempted previously.

In this paper, we introduce *BALF*, a simple yet both efficient and effective motion blur aware local feature detector. Our detector network consists of two main components: a pure MLP-based image encoder and a keypoint detection module. The encoder takes advantage of a cascaded of multi-axis gated MLP blocks and "Squeeze and Excitation (SE)" MLP blocks, to learn a pyramid feature representation of the input image. The keypoint detection module then apply a differentiable channel-wise softmax operator to detect salient keypoints. Extensive evaluations have been conducted on both synthetic and real datasets. Experimental results demonstrate that our method achieves superior performance over prior methods as shown in Fig. 1. In particular, we achieve 15% improvements over the current best performing network on motion blurred images with the repeatability metric, which is commonly used for local feature detection evaluations. Besides the repeatability performance, our method is also able to run at around 35 FPS for VGA resolution image (*i.e*. 480×640 pixels) on a consumer-grade Graphic Card (*i.e*. NVIDIA Geforce 2080 Ti), which is sufficient for time-constrained applications.

In summary, our **contributions** are as follows:

- We propose a novel and efficient MLP-based network architecture for local feature detection, which has never been attempted for this task previously.
- Extensive experimental results demonstrate that our network achieves superior detection performance over prior works on motion blurred images, while keeping comparable performance for sharp images.
- Our motion blur robust keypoint detector is able to run in real-time, which would enable many time-constrained applications (*e.g*. robotic navigation in low-lighting scenarios).

2. Related Work

We review three main areas of related work: keypoint detection, image deblurring, and MLP related methods.

Keypoint detection. Local feature detection plays a vital role in many vision related tasks, such as visual localization and recognition. It thus received a continuous influx of attention in the past decades [11, 15, 50]. Existing works can be generally categorized into classical handcrafted based methods and modern learning based methods. Since our work belongs to learning based approach, we pay our attention on reviewing learned feature detectors. For more details on classical handcrafted feature detectors, interested readers can refer to a benchmark work from Schmid *et al*. [52].

FAST [46] is one of the first attemps to apply machine learning technique for reliable and fast corner detection. Similar strategies have also been applied in other related extensions [27, 47, 48]. TILDE [65], one of the first deep learning based detector, trains a piece-wise linear regressor to detect keypoints under drastic weather and illuminstion changes, using SIFT keypoints as supervision. DetNet [26] derives a covariant constraint to learn stable anchors for local features. It is further extended by Zhang *et al*. [79] to introduce two new concepts of standard patch and canonical feature for feature detection. Savinov *et al*. [51] later proposes Quad-networks, which is unsupervised and trains a neural network to rank points under a transformation-invaraint manner. It then extracts keypoints from the top/bottom quantiles of the rankings. A similar detector is proposed by Zhang *et al*. [78], which combines the same ranking loss with a grid-wise peakness loss to detect keypoints in texture images. Key.Net [25] resorts to using hand-crafted filters together with learned convolutional neural network (CNN) features by a light weight CNN network. They propose to use a spatial softmax operator for detecting keypoints across multi-scale regions. ELF [7] proposes to use a pre-trained CNN for image classification task to detect saliency keypoints without requiring extra training. Recently, it has been shown that convolutional neural network trained for descriptor can also be used for keypoint detection, and achieves impressive performance [61]. In addition to the above methods which are solely designed for keypoint detection, existing works also seek the possibility to integrate both the feature detection and description in a unified framework. For example, LIFT [74] takes a quadruplet of patches to jointly train a detector, an orientation estimator, and a descriptor which are supervised by the feature correspondences from a Structure-from-Motion (SfM) pipeline. Instead of getting supervision via a SfM pipeline, SuperPoint [12] proposes to first pre-train the detector via a synthetic dataset, which consists of primitive geometric shapes. They then take advantage of a homographic adaptation module to achieve self-supervised training together with the feature descriptor network on real images. LF-Net [41] proposes to enforce same feature response for corresponding points across images to train the detector. The correspondence is built based on known camera poses and depth maps.

Different from existing works, which are usually trained

to detect keypoints from a sharp image, we propose a motion blur aware keypoint detector for robust 3D vision under low-lighting conditions.

Image deblurring. Existing motion deblurring algorithms can be mainly categorized into classical gradient-descent based methods and learning based methods during inference. We will only focus on several representative learning based single image deblurring networks, which are most related to our work. Early learning based methods [17, 53, 57, 73] mainly exploit deep network to estimate the unknown blur kernels and then employ conventional deconvolution methods to restore the blurred images. The performance of single image deblurring algorithms has been further boosted by end-to-end deep neural networks, which formulate the deblurring problem as an image translation problem. Xu *et al*. [70] develop a deep convolutional neural netowrk to capture the blur degradation for image deconvolution. Sun *et al*. [57] apply the network to predict the varying motion blur kernels, which enables to image deblurring. Nah *et al*. [40] follow a coarse-to-fine approach to train a multi-scale CNN for blind deblurring. Kupyn *et al*. propose DeblurGAN [23] and DeblurGAN-v2 [24] based on a adversarial loss for motion deblurring. Tao *et al*. [60] propose a SRN-DeblurNet, which adopts a scale-recurrent structure to realize multi-scale image deblurring. Liu *et al*. [33] recently propose a self-supervised network for motion deblurring. Although those methods achieve impressive performance on image deblurring, they usually require large computational resources and are hard to run in real-time. Instead of firstly deblurring each image and then detecting keypoints on the restored image, we propose an efficient one-stage feature detector from motion blurred image directly.

MLP-like architecture in computer vision. While convolutional neural networks and Vision Transformers (ViT) [13] have been the de-facto standards for many computer vision applications, MLP-based architectures have also achieved state-of-the-art performance in several vision tasks recently [8, 10, 18, 19, 29, 30, 32, 35, 59, 62–64, 75]. Due to the conceptually and technically simplicity, MLP-based architecture is getting more attention in both visual recognition [32, 75] and dense prediction tasks [8, 30]. Recently, MAXIM [64] adopts a multi-axis gated MLP module for low-level image processing while SegFormer [68] unifies Transformers with MLP decoders for semantic segmentation tasks. While MLP-based architecture has achieved impressive performance in several vision related tasks, their applicability in local feature detection has never been explored. We thus propose to explore this possibility for local feature detection, and achieve state-of-the-art performance over prior keypoint detectors with both motion blurred and natural sharp images.

3. Methods

We present, to the best of our knowledge, the first MLP-based architecture (as shown in Fig. 2) for local feature detection with a blurred image. As shown in Fig. 2a, our network consists of a MLP-based encoder and a MLP-based detection module. The encoder network learns a effective feature representation of the input image via cascaded MLP blocks. The learned feature representation is then input to a detection module, which takes advantages of a differentiable channel-wise softmax operator. The resulting network is light-weight and effective. We will detail each component as follows.

3.1. MLP-based encoder

Our MLP-based encoder contains cascaded MLPCoder blocks (as shown in Fig. 2b) to obtain a pyramid level representation of the input image. Each MLPCoder block contains a channel MLP block, a multi-axis gated MLP block and a residual MLP attention block. The channel MLP block maps each pixel to a high-dimensional representation, such that it can be further processed by the following blocks. After each MLPCoder block, we apply a max-pooling layer to extract the most significant features and reduce the spatial dimension of the feature representation. The loss of the spatial resolution caused by the pooling layer is compensated by the increased number of channels of the feature representation.

Multi-axis gated MLP block. For keypoint detection task, local region relationship among the pixels is usually more important compared to long-range relationship. However, due to motion blur, the intensity of a particular pixel is mixed by that of several neighboring pixels. It would be beneficial to have a larger receptive field to take account of more context information for better localization of the salient keypoints. We thus take advantages of a state-of-the-art MLP block, *i.e*. the multi-axis gated MLP block, from MAXIM [64] to extract both local and global visual cues. MAXIM [64] is a MLP-based network for low-level image processing tasks, and achieves state-of-the-art performance on image denoising, deblurring, deraining, dehazing and enhancement compared to prior works. Multi-axis gated MLP block presents a principled way to apply 1D operators on 2D images in a scalable manner, and apply them in parallel corresponds to both local and global (dilated) mixing of spatial information respectively. For more properties and complexity analysis, interested readers are suggested to refer to the work of Tu *et al*. [64].

Residual MLP attention block. The multi-axis gated MLP block mainly learns spatial dependencies across the feature representations. To better capture the channel-wise dependencies and inspired by convolutional channel attention block in [66, 76], we exploit the squeeze-and-excitation

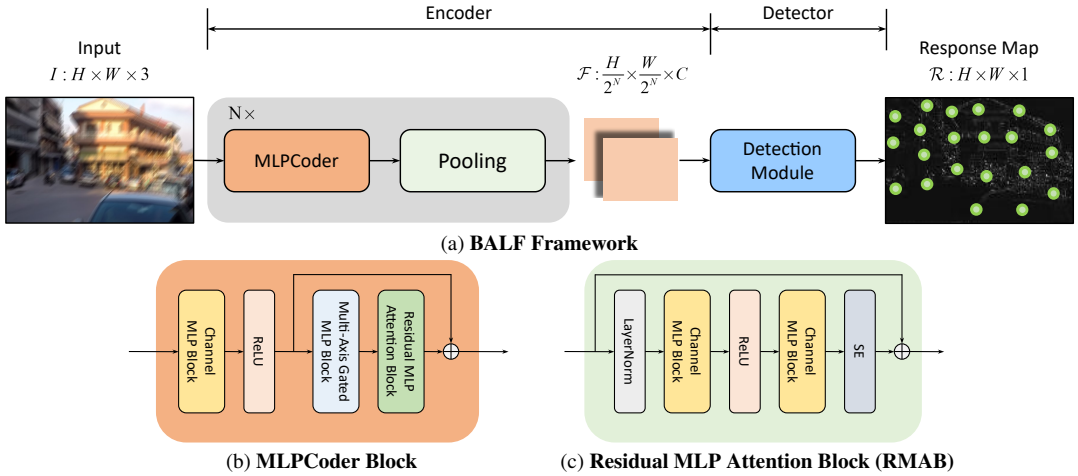

Figure 2. **The proposed network for motion blur aware local feature detector (BALF).** Our network consists of two main modules: a multi-stage MLP-based encoder to extract an intermediate feature representation of the input image, and a MLP detection module to detect salient keypoints via a differentiable softmax operator.

(SE) block from SENet [21] to build up a residual MLP attention block (RMAB). SE block is more efficient compared to those convolutional channel attention blocks. It allows the network to perform feature re-calibration and exploit the inter-channel relationship of features, through which it can learn to use global information to selectively emphasize informative features and suppress less useful ones.

The SE block first conduct a squeeze operation, which produces a channel descriptor by aggregating feature maps across their spatial dimensions. The function of this descriptor is to produce an embedding of the global distribution of channel-wise feature responses, allowing information from the global receptive field of the network to be used by all its layers. The aggregation is followed by an excitation operation, which takes the form of a simple self-gating mechanism that takes the embedding as input and produces a collection of per-channel modulation weights. These weights are applied to the input feature maps to generate the output of the SE block which can be fed directly into subsequent layers of the network. We further improve the channel attention concept for keypoint detection task, by applying two channel MLP blocks to build a residual MLP attention block (as shown in Fig. 2c) together with the SE block [21].

3.2. MLP-based detection module

After the MLP-based encoder, the resulting feature map $\mathcal{F} \in \mathbb{R}^{H' \times W' \times C}$, where $H' = \frac{H}{2^N}$ and $W' = \frac{W}{2^N}$, are passed through a MLP-based detection module (as shown in Fig. 3) to output a dense probability map for each pixel as a keypoint.

The details of the detection module are illustrated in Fig. 3. It consists of two channel MLP blocks, which transform the input feature representation from C channels to C_r channels at the same spatial resolution, where $C_r = 4^N$. Each element along the channel direction corresponds to the response of a particular pixel of the input full resolution image. In other words, all the elements along the channel direction correspond to the responses of pixels within a $2^N \times 2^N$ sized patch of the input image. We use $N = 3$ in our experiments.

Handcrafted feature detectors, such as SIFT [34], usually perform spatial non-maxima suppression (NMS) on the response map, to select a set of sparsely distributed salient keypoints. To achieve similar goal, we apply a differentiable channel-wise softmax operation on the transformed feature representation as shown in Fig. 3. The corresponding channel index of the maximum responsed element is then mapped to the pixel index of the full resolution input image, such that the corresponding pixel can be assigned with the response score. We further eliminate pixels as keypoints for those with low responses by a pre-defined threshold, such as pixels within a homogeneous region. The purpose of using this feature detection procedure is two-fold. First, the channel-wise softmax layer is similar to NMS but is differentiable and enables end-to-end training. Secondly, it replaces the decoder module with simple non-parametric operator. It neither involves any feature learning nor introduces any additional parameters, which further reduces the computational overhead.

3.3. Loss function

We formulate the keypoint detection as a regression problem. We train the network with a ground truth response map $\mathcal{R}_{GT}$, which are generated by detecting SIFT keypoints [34] on the corresponding sharp images and placing Gaussian kernels at those locations. The loss function

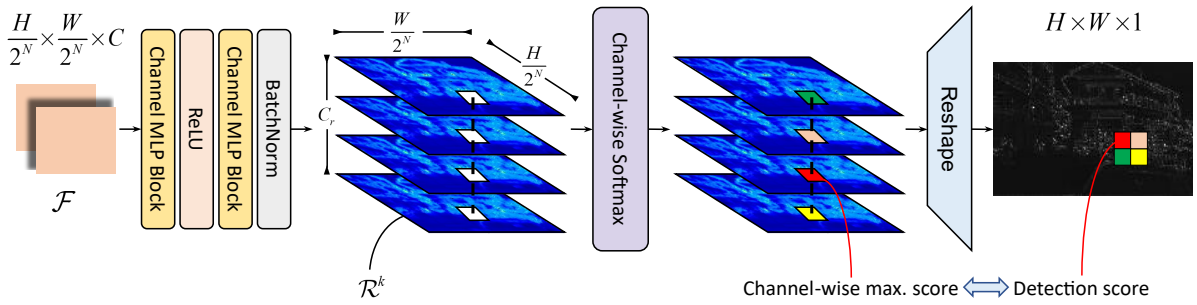

Figure 3. **Detection module.** The learned feature representations of the input image is processed by two channel-wise MLP blocks. The keypoints are then detected by using channel-wise softmax operation and mapped back to the original image domain.

used to train the network can be formalized described as:

$$\mathcal{L} = ||\mathcal{R} - \mathcal{R}_{GT}||^2, \tag{1}$$

where $\mathcal{R}$ is the predicted response map (as shown in Fig. 2a) and $\mathcal{R}_{GT}$ is the ground truth response map.

4. Experiments

The main motivation of our work is to develop a motion blur aware local feature detector via deep networks, since many state-of-the-art methods for a sharp image have been proposed and achieved impressive performance. There is no existing dataset to evaluate local feature detectors on motion blurred images. We thus create a training dataset via a publicly available single image deblurring dataset, *i.e*. the GoPro dataset from Nah *et al*. [40], which has paired sharp and blurred images. To evaluate our method, we generate a synthetic dataset via HPatches dataset [5], which is commonly used for local feature evaluations. HPatches dataset is different from the GoPro dataset, and it can thus also reflect the generalization performance of our method. We therefore mainly use this dataset for evaluation.

Datasets. We use the GoPro dataset from Nah *et al*. [40] to generate data to train our network. The GoPro dataset is commonly used for evaluations of single image deblurring networks. It consists of 3,214 paired sharp and blurred images, which are captured from 33 scenes. We follow their convention and use 22 sequences for training and 11 sequences for testing. To generate supervision data for each blurred image, we assume the blurred image should have the same keypoint locations as those for its paired sharp image. We use SIFT [34] to first detect keypoints from the sharp image, and then generate a heatmap and place Gaussian kernels as the ground truth response map for network training.

To better evaluate the generalization performance of our method, we generate a synthetic motion blurred image dataset by using HPatches dataset [5]. HPatches dataset is commonly used for local feature evaluations, which originally covers both illumination and viewpoint changes. It

Figure 4. **Example synthesized blurred images from HPatches dataset [5].** The first column shows the sharp image, and the next three columns are blurred images with varying blur levels. Best viewed in high resolution.

gathers images from existing datasets, such as DTU [1] and Oxford [39] datasets. It provides a total of 116 sequences and is further divided into 59 sequences for viewpoint changes and 57 sequences for illumination changes. Each sequence includes a reference image and 5 target images with varying viewpoint or illumination changes, together with the corresponding homographies between them. We generate a random motion blur kernel, *i.e*. point spread function, for each image from the HPatches dataset to synthesize a motion blurred image. We generate three varying levels of motion blur, by changing the blur kernel size and motion irregularities (*i.e*. the non-linearity of the motion). Fig. 4 presents several example images for different blur levels. We use this dataset purely for evaluation.

Implementation details. Our network is light-weight and end-to-end trainable. It requires neither large-scale pretraining nor progressive training. During training, we use a batch size of 4, and Adam optimizer with a initial learning rate of 10^{-4}. After the network is trained for 20 epochs, we linearly decay the learning rate to 10^{-6} till the 50^{th} epoch. The network takes about 3 hours to be trained on a single NVIDIA Geforce 2080Ti graphic card. During training, we also perform data augmentation as that of [25]. In particular, we perform a sequence of random rotation, random scaling, random skewing and random perspective transformation on the original image. We also apply random pho-

Variant	Repeatability ↑	Params ↓	Inference time ↓
w/o RMAB	63.59%	326K	19.18ms
RCAB	73.56%	725K	57.29ms
RMAB (proposed)	75.15%	381K	29.02ms

Table 1. **Ablation study of the residual MLP attention block (RMAB).** The experimental results demonstrate that the RMAB block is indeed efficient and effective for local feature detection.

tometric transformation, such that the trained network is robust against illumination changes. Then we randomly crop a 256×256 pixels image for training. Detailed network architecture can be found in our supplementary material.

Baseline methods and evaluation metrics. We compare our approach against a number of representative detectors, which range from classical handcrafted methods to recent learning based methods. In particular, we compare against SIFT [34], SURF [6], Harris Laplace [38], Shi-Tomasi [56], MSER [36], KAZE [2], AKAZE [3] and FAST [47] with their OpenCV implementations. We also compare with learned feature detectors, such as LIFT [74], Key.Net [25], SuperPoint [12], LF-Net [41], D2-Net [14], and R2D2 [44]. For fair comparisons with motion blurred images, we also re-train those learning based methods on the GoPro dataset using their publicly available source codes. For evaluations with sharp images, we still use their official provided pre-trained models.

The repeatability metric proposed in [37] measures the quality of keypoint detection and is commonly used by the community. For a pair of images, it is computed as the ratio between the number of corresponding keypoints observed by both images and the smaller number of keypoints detected in one of the two images. To identify the corresponding keypoints, we compute the overlap error, ϵ_{IoU}, between the regions of two candidate keypoints as in [78, 79]. We also fix the maximum number of extracted keypoints and allow each keypoint to be matched only once as in [74] for fair evaluations. In our experiments, we consider the top 1000 keypoints for repeatability computation. It is considered as a correct match if ϵ_{IoU} is smaller than 0.4, *i.e.*, the overlap between corresponding region is more than 60%.

Ablation study In this section, we study the effect of the residual MLP attention block (RMAB). We conduct experiments with three settings, *i.e.* the network with and without RMAB, and we also replace RMAB with a residual convolutional attention block (RCAB) from [66]. The experimental results shown in Tab. 1 demonstrate that RMAB block is indeed effective for local feature detection. The RCAB block requires more parameters, longer inference time and achieves lower repeatability compared to the network with RMAB. It further demonstrates the potential to use MLP blocks for efficient network architectures. The experiments are conducted with the GoPro testing sequences.

Evaluation with the original HPatches dataset. To study the performance of our method on sharp images, we compare it against other methods on the original HPatches dataset [5]. For this experiment, our network is trained on the GoPro dataset with both sharp and blurred images, while we use the official pre-trained netwroks for other methods. Those pre-trained models are usually well trained with a much larger dataset, such as the COCO dataset [31] or the ImageNet dataset [49]. The experimental results are presented in Tab. 2. It can be demonstrated that our method performs on-par with the state-of-the-art method (*i.e.* LF-Net [41]) for local feature detection with sharp images. Our method is only slightly worse (*i.e.* $\sim$0.7% drop with the repeatability metric) compared to LF-Net [41], although our network is trained with a relatively small dataset which contains both sharp and blurred images. The experimental results further demonstrate that our detector (*i.e.* BALF) has superior performance even it is designed and trained to be robust against motion blur.

Evaluation with the Blur-HPatches dataset. To evaluate the performance of our network with motion blurred images, we evaluate it against other methods on the synthesized blurred HPatches dataset. For fair comparisons, we re-train all learning based methods on the same GoPro dataset. For compactness, we report the total repeatability instead of the separated results for both viewpoint and illumination changes as in Tab. 3. Detailed results on the respective changes can be found in our supplementary material. We also simulate two different application scenarios, *i.e.* the evaluations with blur-to-sharp and blur-to-blur configurations. The blur-to-sharp configuration can be applied for visual localization task. For example, we can pre-built a large-scale 3D map with high-quality sharp images. It might happen that we would use blurred image to query its location against the 3D map if we walk around with an AR device at night. The blur-to-blur configuration can be applied for visual odometry task, from which all the captured images within a time window are motion blurred.

The experimental results shown in Tab. 3 demonstrate that prior methods have degraded detection performance when the images are motion blurred. The performance degrades more as the motion blur becomes severer. However, our method achieves impressive performance compared to prior works. The reason might be that prior learning based methods are usually built based on simple convolutional layers (*e.g.* SuperPoint [12]) or networks which are not specially designed for image deblurring. Motion blurred images thus challenge those networks on keypoint detection task. In contrary, our network is built based on the multi-axis gated MLP block, which has been demonstrated to be effective for low-level image processing, *e.g.* image deblurring [64].

Evaluation with the GoPro dataset. We also evaluate

Method	Reference: Sharp Target: Sharp		
	Viewpoint ↑	Illumination ↑	Total ↑
SIFT [34]	60.29	60.44	60.36
SURF [6]	62.67	64.01	63.33
Harris-Laplace [38]	63.89	62.91	63.41
Shi-Tomasi [56]	69.28	64.13	66.74
MSER [36]	52.45	50.58	51.53
KAZE [2]	67.30	65.67	66.50
AKAZE [3]	66.08	69.07	67.55
FAST [47]	66.08	63.65	64.88
LIFT [74]	56.97	60.73	58.82
Key.Net [25]	68.99	67.47	68.24
SuperPoint [12]	**69.53**	68.92	69.23
LF-Net [41]	68.41	**73.61**	**70.96**
D2-Net [14]	53.99	62.80	58.32
R2D2 [44]	61.68	61.93	61.80
BALF (ours)	67.21	73.51	70.28

Table 2. **Repeatability results (%) on HPatches [5] sharp image pairs.** For each method, we report the average repeatability score on the viewpoint change, illumination change, and all image sequences.

Method	Reference: Sharp Target: Blur			Reference: Blur Target: Blur		
	EASY ↑	HARD ↑	TOUGH ↑	EASY ↑	HARD ↑	TOUGH ↑
SIFT [34]	55.92	56.80	53.49	56.99	53.49	45.94
SURF [6]	58.88	56.23	56.24	61.08	58.04	53.60
Harris-Laplace [38]	36.70	37.97	34.98	35.76	31.95	27.47
Shi-Tomasi [56]	57.33	55.11	49.11	56.29	53.75	51.37
MSER [36]	44.19	41.97	37.05	41.81	38.24	34.59
KAZE [2]	49.90	46.84	39.98	63.29	58.71	46.90
AKAZE [3]	54.15	50.51	45.49	65.16	62.20	51.54
FAST [47]	61.98	61.77	51.37	57.84	53.35	51.17
LIFT [74]	50.69	50.17	46.99	48.34	46.57	46.53
Key.Net [25]	60.34	54.71	44.69	62.77	58.17	49.25
SuperPoint [12]	65.64	62.22	52.84	58.60	50.03	43.28
LF-Net [41]	63.54	61.19	56.78	60.45	59.07	57.71
D2-Net [14]	49.71	47.30	44.32	51.80	51.05	50.53
R2D2 [44]	57.99	51.73	40.57	57.49	55.31	46.86
BALF (ours)	**74.12**	**74.45**	**71.84**	**70.48**	**68.43**	**67.71**

Table 3. **Repeatability results (%) on Blur-HPatches datasets.** Our method achieves the best performance on both blur-to-sharp and blur-to-blur configurations. For compactness, we only report the total repeatability. Detailed results on the respective changes can be found in our supplementary material.

Method	Reference: Sharp Target: Deblur						Reference: Deblur Target: Deblur					
	SRN-DeblurNet [60]			DeblurGAN-v2 [24]			SRN-DeblurNet [60]			DeblurGAN-v2 [24]		
	EASY ↑	HARD ↑	TOUGH ↑	EASY ↑	HARD ↑	TOUGH ↑	EASY ↑	HARD ↑	TOUGH ↑	EASY ↑	HARD ↑	TOUGH ↑
SIFT [34]	56.62	55.36	53.83	57.63	56.52	56.50	59.75	58.13	50.63	59.44	57.98	51.21
SURF [6]	61.89	59.13	54.88	61.97	59.57	56.34	62.44	61.26	55.27	62.07	60.81	55.09
Harris-Laplace [38]	17.15	16.87	20.54	16.60	16.90	20.24	36.98	35.73	32.23	37.09	35.97	31.54
Shi-Tomasi [56]	60.56	56.87	48.78	61.75	59.10	51.56	63.18	61.03	50.88	63.58	61.89	53.76
MSER [36]	46.65	43.23	37.90	47.62	45.14	40.70	47.70	45.40	37.49	47.84	45.56	38.01
KAZE [2]	65.14	63.10	60.16	65.23	63.18	61.41	64.20	62.45	53.41	64.13	61.87	54.19
AKAZE [3]	66.03	64.02	60.64	66.29	64.50	62.72	65.71	64.08	56.10	65.75	63.75	57.35
FAST [47]	61.77	59.67	61.60	62.00	60.44	58.74	62.72	61.14	50.61	63.40	61.70	55.43
LIFT [74]	54.98	52.64	46.75	56.59	53.54	49.09	55.88	53.64	45.31	56.68	55.31	50.44
Key.Net [25]	63.28	58.01	47.10	63.99	59.16	49.35	62.86	60.44	50.74	62.73	60.58	52.96
SuperPoint [12]	67.72	64.05	55.26	67.95	65.86	58.22	66.38	63.16	49.52	66.50	63.71	52.09
LF-Net [41]	62.22	59.90	54.73	62.59	60.24	54.81	63.06	62.03	57.28	63.00	61.79	57.85
D2-Net [14]	51.81	49.49	45.94	52.64	50.21	45.88	53.60	53.00	50.93	53.93	53.29	50.74
R2D2 [44]	60.31	55.43	43.26	60.46	55.68	45.38	58.11	54.80	45.77	57.95	55.03	47.86
BALF (ours)	**74.12 / 74.45 / 71.84** (EASY / HARD / TOUGH)						**70.48 / 68.43 / 67.71** (EASY / HARD / TOUGH)					

Table 4. **Repeatability results (%) on deblurred images.** The experimental results demonstrate that single image deblurring network can indeed help with the local feature detection. However, it still cannot perform on-par with our one-stage detection network without doing any intermediate deblurring operation.

the performance of our network against the other methods on the GoPro testing sequences. For fair comparisons, all learning based methods are re-trained on the GoPro dataset. Since there is no ground truth homographies for GoPro dataset as those of HPatches dataset [5], we randomly warp each testing image to get paired transformed image for repeatability evaluation. The experimental results presented in Tab. 5 demonstrate that our network achieves superior performance compared to other approaches.

Evaluation with the Blur-HPatches dataset preprocessed by deblurring network. We also study if the deblurring networks can help with keypoint detection from blurred image even they usually cannot run in real-time. In particular, we apply two state-of-the-art deblurring networks, *i.e*. SRN-DeblurNet [60] and DeblurGAN-v2 [24], to deblur the images from the Blur-HPatches dataset and then evaluate all the other methods on the restored images. We use the official pretrained models and apply them to images from the Blur-HPatches dataset preprocessed by these two deblurring networks without any finetuning.

The experimental results shown in Tab. 4 demonstrate that the deblurring networks could indeed help a bit for key-

Method	Reference: Sharp Target: Blur ↑	Reference: Blur Target: Blur ↑
SIFT [34]	60.53	60.03
SURF [6]	56.49	60.03
Harris-Laplace [38]	23.54	16.35
Shi-Tomasi [56]	51.26	57.61
MSER [36]	46.54	43.09
KAZE [2]	49.35	48.86
AKAZE [3]	56.34	50.51
FAST [47]	51.20	45.04
LIFT [74]	48.61	50.56
Key.Net [25]	57.54	58.37
SuperPoint [12]	53.83	51.38
LF-Net [41]	60.82	66.60
D2-Net [14]	53.37	56.96
R2D2 [44]	50.34	53.94
BALF (ours)	**75.68**	**75.15**

Table 5. **Repeatability results (%) on GoPro testing dataset.** The experimental results demonstrate that our network also achieves the state-of-the-art performance compared to prior works on the GoPro dataset.

point detection from blurred image. For example, it improves SuperPoint [12] from 62.22% to 64.05% on the repeatability metric for the hard blur-to-sharp case. However, it still cannot outperform our network, which has 74.45% repeatability on the blurred image directly. The reason might be that single image deblurring networks have limitations to restore image from severe motion blurred image. It further demonstrates that to design an one-stage keypoint detector from blurred image directly, would be a better option compared to that of detecting keypoints from the intermediate deblurred image.

Efficiency and performance with real blurred images. We also evaluate the efficiency of our method against other methods. Tab. 6 presents the computational time for all those keypoint detectors. The handcrafted detectors are evaluated on a CPU (Intel i7-8700), and remaining learning based methods[1] run on a NVIDIA Geforce 2080 Ti. For fair comparisons, we remove the descriptor network for evaluation from LIFT, SuperPoint, and LF-Net. The experimental results demonstrate that our motion blur aware detector is able to run in real-time (∼34.46 FPS) with a VGA resolution image (*i.e.* 480×640 pixels). It further demonstrates that our detector can be applied to many time-constrained applications, such as robotic visual navigation.

To further demonstrate the performance of our network on the real blurred images, we also present a qualitative feature matching result between a sharp image and a blurred image in Fig. 5. The images are selected from the RealBlur dataset [45], which are captured by real cameras. It demonstrates that our network is able to detect well localized and repeatable keypoints from both sharp and blurred images. More details on the matching implementation and qualitative results can be found in our supplementary material.

[1]Since it is impossible to separate the detector and descriptor networks for individual evaluations from D2-Net or R2D2, we did not measure their time consumption.

Figure 5. **Qualitative evaluations on real blurred image.** The experimental results demonstrate that our network is able to detect well distributed and localized keypoints from either sharp and blurred images for further image matching.

Method	240×320 pixels ↓	480×640 pixels ↓
SIFT [34]	21.80	66.70
SURF [6]	148.46	165.78
Harris-LapLace [38]	110.41	377.13
Shi-Tomasi [56]	5.20	21.69
MSER [36]	64.19	221.79
KAZE [2]	105.85	298.43
AKZE [3]	18.02	56.93
FAST [47]	0.89	1.88
LIFT [74]	2209.03	4901.38
Key.Net [25]	15.64	35.82
SuperPoint [12]	2.41	3.98
LF-Net [41]	282.77	855.77
BALF (ours)	8.15	29.02

Table 6. **Computational cost (in millisecond)** for different keypoint detectors for a single image. It demonstrates that our network can run in real-time for a VGA resolution image, *i.e.* 480×640 pixels, which is commonly used for many robotic applications.

5. Conclusion and Future Works

We present the first pure MLP-based network for local feature detection. Our network takes advantages of the multi-axis gated MLP block and a squeeze-and-excitation MLP block to build a pure MLP-based image encoder. The detection module then applies differentiable channel-wise softmax operator for keypoint detection. Extensive evaluations have been conducted with both synthetic and real datasets. The experimental results demonstrate that our network delivers on-par detection performance on sharp images, and achieves superior performance with motion blurred images compared to prior works. Our network is also light-weight and is able to run in real-time for VGA resolution image, which further enables its application for time-constrained applications. It is usually required to build correspondences via feature matching for higher level applications, such as visual localization. We thus plan to design a network to learn motion blur robust descriptors for those detected keypoints as our future work.

References

[1] Henrik Aanæs, A. Dahl, and Kim Steenstrup Pedersen. Interesting interest points. *IJCV*, 97:18–35, 2011. 5

[2] Pablo Fernández Alcantarilla, Adrien Bartoli, and Andrew J. Davison. Kaze features. In *ECCV*, 2012. 6, 7, 8

[3] Pablo Fernández Alcantarilla, Jesús Nuevo, and Adrien Bartoli. Fast explicit diffusion for accelerated features in nonlinear scale spaces. In *BMVC*, 2013. 6, 7, 8

[4] Yuval Bahat, Netalee Efrat, and Michal Irani. Non-uniform blind deblurring by reblurring. In *ICCV*, pages 3306–3314, 2017. 1

[5] Vassileios Balntas, Karel Lenc, Andrea Vedaldi, and Krystian Mikolajczyk. Hpatches: A benchmark and evaluation of handcrafted and learned local descriptors. In *CVPR*, pages 3852–3861, 2017. 5, 6, 7

[6] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. Surf: Speeded up robust features. In *ECCV*, 2006. 6, 7, 8

[7] Assia Benbihi, Matthieu Geist, and Cédric Pradalier. Elf: Embedded localisation of features in pre-trained cnn. In *ICCV*, pages 7939–7948, 2019. 1, 2

[8] Shoufa Chen, Enze Xie, Chongjian Ge, Ding Liang, and Ping Luo. Cyclemlp: A mlp-like architecture for dense prediction. In *ICLR*, 2022. 2, 3

[9] Sunghyun Cho and Seungyong Lee. Fast motion deblurring. In *ACM SIGGRAPH Asia*, 2009. 1

[10] Jaesung Choe, Chunghyun Park, François Rameau, Jaesik Park, and In-So Kweon. Pointmixer: Mlp-mixer for point cloud understanding. In *ECCV*, 2022. 2, 3

[11] Gabriela Csurka and M. Humenberger. From handcrafted to deep local invariant features. *ArXiv*, abs/1807.10254, 2018. 2

[12] Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superpoint: Self-supervised interest point detection and description. In *CVPRW*, pages 337–33712, 2018. 2, 6, 7, 8

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 3

[14] Mihai Dusmanu, Ignacio Rocco, Tomás Pajdla, Marc Pollefeys, Josef Sivic, Akihiko Torii, and Torsten Sattler. D2-net: A trainable cnn for joint description and detection of local features. In *CVPR*, pages 8084–8093, 2019. 6, 7, 8

[15] Steffen Gauglitz, Tobias Höllerer, and Matthew A. Turk. Evaluation of interest point detectors and feature descriptors for visual tracking. *IJCV*, 94:335–360, 2011. 2

[16] Amit Goldstein and Raanan Fattal. Blur-kernel estimation from spectral irregularities. In *ECCV*, 2012. 1

[17] Dong Gong, Jie Yang, Lingqiao Liu, Yanning Zhang, Ian D. Reid, Chunhua Shen, Anton van den Hengel, and Qinfeng Shi. From motion blur to motion flow: A deep learning solution for removing heterogeneous motion blur. In *CVPR*, pages 3806–3815, 2017. 3

[18] Jianyuan Guo, Yehui Tang, Kai Han, Xinghao Chen, Han Wu, Chao Xu, Chang Xu, and Yunhe Wang. Hire-mlp: Vision mlp via hierarchical rearrangement. In *CVPR*, pages 816–826, 2022. 2, 3

[19] Qibin Hou, Zihang Jiang, Li Yuan, Mingg-Ming Cheng, Shuicheng Yan, and Jiashi Feng. Vision permutator: A permutable mlp-like architecture for visual recognition. *IEEE TPAMI*, 2022. 2, 3

[20] Michal Hradis, Jan Kotera, Pavel Zemcik, and Filip Sroubek. Convolutional neural networks for direct text deblurring. In *BMVC*, 2015. 1

[21] Jie Hu, Li Shen, Samuel Albanie, Gang Sun, and Enhua Wu. Squeeze-and-excitation networks. *IEEE TPAMI*, 42:2011–2023, 2020. 4

[22] Dilip Krishnan, Terence Tay, and Rob Fergus. Blind deconvolution using a normalized sparsity measure. In *CVPR*, pages 233–240, 2011. 1

[23] Orest Kupyn, Volodymyr Budzan, Mykola Mykhailych, Dmytro Mishkin, and Jiri Matas. Deblurgan: Blind motion deblurring using conditional adversarial networks. In *CVPR*, pages 8183–8192, 2018. 1, 3

[24] Orest Kupyn, T. Martyniuk, Junru Wu, and Zhangyang Wang. Deblurgan-v2: Deblurring (orders-of-magnitude) faster and better. In *ICCV*, pages 8877–8886, 2019. 1, 3, 7

[25] Axel Barroso Laguna, Edgar Riba, Daniel Ponsa, and Krystian Mikolajczyk. Key.net: Keypoint detection by handcrafted and learned cnn filters. In *ICCV*, pages 5835–5843, 2019. 1, 2, 5, 6, 7, 8

[26] Karel Lenc and Andrea Vedaldi. Learning covariant feature detectors. In *ECCV Workshops*, 2016. 2

[27] Stefan Leutenegger, Margarita Chli, and Roland Y. Siegwart. Brisk: Binary robust invariant scalable keypoints. In *ICCV*, pages 2548–2555, 2011. 2

[28] Lerenhan Li, Jinshan Pan, Wei-Sheng Lai, Changxin Gao, Nong Sang, and Ming-Hsuan Yang. Learning a discriminative prior for blind image deblurring. In *CVPR*, pages 6616–6625, 2018. 1

[29] Wenshuo Li, Hanting Chen, Jianyuan Guo, Ziyang Zhang, and Yunhe Wang. Brain-inspired multilayer perceptron with spiking neurons. In *CVPR*, pages 773–783, 2022. 2, 3

[30] Dongze Lian, Zehao Yu, Xing Sun, and Shenghua Gao. As-mlp: An axial shifted mlp architecture for vision. In *ICLR*, 2022. 2, 3

[31] 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 *ECCV*, 2014. 6

[32] Hanxiao Liu, Zihang Dai, David R. So, and Quoc V. Le. Pay attention to mlps. In *NeurIPS*, 2021. 2, 3

[33] Peidong Liu, Joel Janai, Marc Pollefeys, Torsten Sattler, and Andreas Geiger. Self-supervised linear motion deblurring. *IEEE Robotics and Automation Letters*, 5:2475–2482, 2020. 1, 3

[34] G LoweDavid. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004. 4, 5, 6, 7, 8

[35] Youssef Mansour, Kang Lin, and Reinhard Heckel. Image-to-image mlp-mixer for image reconstruction. *ArXiv*, abs/2202.02018, 2022. 2, 3

[36] Jiri Matas, Ondřej Chum, Martin Urban, and Tomás Pajdla. Robust wide-baseline stereo from maximally stable extremal regions. *Image Vis. Comput.*, 22:761–767, 2004. 6, 7, 8

[37] Krystian Mikolajczyk and Cordelia Schmid. A performance evaluation of local descriptors. In *CVPR*, pages II–II, 2003. 6

[38] Krystian Mikolajczyk and Cordelia Schmid. Scale & affine invariant interest point detectors. *IJCV*, 60:63–86, 2004. 6, 7, 8

[39] Krystian Mikolajczyk, Tinne Tuytelaars, Cordelia Schmid, Andrew Zisserman, Jiri Matas, Frederik Schaffalitzky, Timor Kadir, and Luc Van Gool. A comparison of affine region detectors. *IJCV*, 65:43–72, 2005. 5

[40] Seungjun Nah, Tae Hyun Kim, and Kyoung Mu Lee. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *CVPR*, pages 257–265, 2017. 1, 3, 5

[41] Yuki Ono, Eduard Trulls, Pascal V. Fua, and Kwang Moo Yi. Lf-net: Learning local features from images. In *NeurIPS*, 2018. 2, 6, 7, 8

[42] Daniel J. Perrone and Paolo Favaro. Total variation blind deconvolution: The devil is in the details. In *CVPR*, pages 2909–2916, 2014. 1

[43] Wenqi Ren, Jiawei Zhang, Lin Ma, Jinshan Pan, Xiaochun Cao, Wangmeng Zuo, W. Liu, and Ming-Hsuan Yang. Deep non-blind deconvolution via generalized low-rank approximation. In *NeurIPS*, 2018. 1

[44] Jérôme Revaud, Philippe Weinzaepfel, César Roberto de Souza, No'e Pion, Gabriela Csurka, Yohann Cabon, and M. Humenberger. R2d2: Repeatable and reliable detector and descriptor. In *NeurIPS*, 2019. 6, 7, 8

[45] Jaesung Rim, Hoon Sung Chwa, and Sunghyun Cho. Real-world blur dataset for learning and benchmarking deblurring algorithms. In *ECCV*, 2020. 8

[46] Edward Rosten and Tom Drummond. Machine learning for high-speed corner detection. In *ECCV*, 2006. 2

[47] Edward Rosten, Reid B. Porter, and Tom Drummond. Faster and better: A machine learning approach to corner detection. *IEEE TPAMI*, 32:105–119, 2010. 2, 6, 7, 8

[48] Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary R. Bradski. Orb: An efficient alternative to sift or surf. In *ICCV*, pages 2564–2571, 2011. 2

[49] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 115(3):211–252, 2015. 6

[50] Ehab Salahat and Murad Qasaimeh. Recent advances in features extraction and description algorithms: A comprehensive survey. In *2017 IEEE International Conference on Industrial Technology (ICIT)*, pages 1059–1063, 2017. 2

[51] Nikolay Savinov, Akihito Seki, Lubor Ladicky, Torsten Sattler, and Marc Pollefeys. Quad-networks: Unsupervised learning to rank for interest point detection. In *CVPR*, pages 3929–3937, 2017. 1, 2

[52] Cordelia Schmid, Roger Mohr, and Christian Bauckhage. Evaluation of interest point detectors. *IJCV*, 37:151–172, 2004. 2

[53] Christian J. Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE TPAMI*, 38:1439–1451, 2016. 1, 3

[54] Jin shan Pan, Zhe Hu, Zhixun Su, and Ming-Hsuan Yang. Deblurring text images via l0-regularized intensity and gradient prior. In *CVPR*, pages 2901–2908, 2014. 1

[55] Jin shan Pan, Deqing Sun, Hanspeter Pfister, and Ming-Hsuan Yang. Blind image deblurring using dark channel prior. In *CVPR*, pages 1628–1636, 2016. 1

[56] Jianbo Shi and Carlo Tomasi. Good features to track. In *CVPR*, pages 593–600, 1994. 6, 7, 8

[57] Jian Sun, Wenfei Cao, Zongben Xu, and Jean Ponce. Learning a convolutional neural network for non-uniform motion blur removal. In *CVPR*, pages 769–777, 2015. 1, 3

[58] Suwichaya Suwanwimolkul, Satoshi Komorita, and Kazuyuki Tasaka. Learning of low-level feature keypoints for accurate and robust detection. pages 2261–2270, 2021. 1

[59] Yehui Tang, Kai Han, Jianyuan Guo, Chang Xu, Yanxi Li, Chao Xu, and Yunhe Wang. An image patch is a wave: Phase-aware vision mlp. In *CVPR*, pages 10925–10934, 2022. 2, 3

[60] Xin Tao, Hongyun Gao, Yi Wang, Xiaoyong Shen, Jue Wang, and Jiaya Jia. Scale-recurrent network for deep image deblurring. In *CVPR*, pages 8174–8182, 2018. 1, 3, 7

[61] Yurun Tian, Vassileios Balntas, Tony Ng, Axel Barroso Laguna, Y. Demiris, and Krystian Mikolajczyk. D2d: Keypoint extraction with describe to detect approach. In *ACCV*, 2020. 1, 2

[62] Ilya O. Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Daniel Keysers, Jakob Uszkoreit, Mario Lucic, and Alexey Dosovitskiy. Mlp-mixer: An all-mlp architecture for vision. In *NeurIPS*, 2021. 2, 3

[63] Hugo Touvron, Piotr Bojanowski, Mathilde Caron, Matthieu Cord, Alaaeldin El-Nouby, Edouard Grave, Gautier Izacard, Armand Joulin, Gabriel Synnaeve, Jakob Verbeek, and Herv'e J'egou. Resmlp: Feedforward networks for image classification with data-efficient training. *IEEE TPAMI*, 2022. 2, 3

[64] Zhengzhong Tu, Hossein Talebi, Han Zhang, Feng Yang, Peyman Milanfar, Alan Conrad Bovik, and Yinxiao Li. Maxim: Multi-axis mlp for image processing. In *CVPR*, pages 5759–5770, 2022. 2, 3, 6

[65] Yannick Verdie, Kwang Moo Yi, Pascal V. Fua, and Vincent Lepetit. Tilde: A temporally invariant learned detector. In *CVPR*, pages 5279–5288, 2015. 1, 2

[66] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In-So Kweon. Cbam: Convolutional block attention module. In *ECCV*, 2018. 3, 6

[67] Lei Xiao, Jue Wang, Wolfgang Heidrich, and Michael Hirsch. Learning high-order filters for efficient blind deconvolution of document photographs. In *ECCV*, 2016. 1

[68] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, José Manuel Álvarez, and Ping Luo. Segformer: Simple and

efficient design for semantic segmentation with transformers. 2021. 3

[69] Li Xu and Jiaya Jia. Two-phase kernel estimation for robust motion deblurring. In *ECCV*, 2010. 1

[70] Li Xu, Jimmy S. J. Ren, Ce Liu, and Jiaya Jia. Deep convolutional neural network for image deconvolution. In *NeurIPS*, 2014. 1, 3

[71] Li Xu, Shicheng Zheng, and Jiaya Jia. Unnatural l0 sparse representation for natural image deblurring. In *CVPR*, pages 1107–1114, 2013. 1

[72] Xiangyu Xu, Jinshan Pan, Yujin Zhang, and Ming-Hsuan Yang. Motion blur kernel estimation via deep learning. *IEEE TIP*, 27:194–205, 2018. 1

[73] Ruomei Yan and Ling Shao. Blind image blur estimation via deep learning. *IEEE TIP*, 25:1910–1921, 2016. 3

[74] Kwang Moo Yi, Eduard Trulls, Vincent Lepetit, and Pascal V. Fua. Lift: Learned invariant feature transform. In *ECCV*, 2016. 2, 6, 7, 8

[75] Tan Yu, Xu Li, Yunfeng Cai, Mingming Sun, and Ping Li. S2-mlpv2: Improved spatial-shift mlp architecture for vision. 2021. 2, 3

[76] Syed Waqas Zamir, Aditya Arora, Salman Hameed Khan, Munawar Hayat, Fahad Shahbaz Khan, Ming-Hsuan Yang, and Ling Shao. Multi-stage progressive image restoration. In *CVPR*, pages 14816–14826, 2021. 3

[77] Jiawei Zhang, Jinshan Pan, Jimmy S. J. Ren, Yibing Song, Linchao Bao, Rynson W. H. Lau, and Ming-Hsuan Yang. Dynamic scene deblurring using spatially variant recurrent neural networks. In *CVPR*, pages 2521–2529, 2018. 1

[78] Linguang Zhang and Szymon M. Rusinkiewicz. Learning to detect features in texture images. In *CVPR*, pages 6325–6333, 2018. 2, 6

[79] Xu Zhang, Felix X. Yu, Svebor Karaman, and Shih-Fu Chang. Learning discriminative and transformation covariant local feature detectors. In *CVPR*, pages 4923–4931, 2017. 1, 2, 6