Vision Transformer with Key-select Routing Attention for Single Image Dehazing

Lihan TONG, Weijia LI, Qingxia YANG, Liyuan CHEN, and Peng CHEN, Nonmembers

SUMMARY We present Ksformer, utilizing Multi-scale Key-select Routing Attention Module (MKRAM) only in the smaller dimensions to reduce computational complexity. To lower the difficulty of training [25], [26], we strengthen the exchange of information between layers and use skip connections at both the feature and image levels.

1. Introduction

Single image dehazing [3]–[5] aims to restore clear, high-quality images from hazy ones, essential for applications like object detection [2] and semantic segmentation [1]. Traditional methods [6], [8], [28] may not give ideal dehazing results because they can’t cover all scenarios [15]. With the rise of deep learning, convolutional neural networks (CNNs) [4], [5], [16] have been widely applied to image dehazing and have achieved good results. However, because CNNs cannot capture long-range dependencies, this limits further improvement in dehazing effects. Recently, Transformers [17], [18], [20], [21] have been widely used in computer vision tasks because they can capture long-range dependencies. However, they have a problem where their computational complexity is proportional to the square of the image resolution. Many efforts [21]–[24] have been made to address this issue by introducing handcrafted sparsity. But the sparsity added by hand doesn’t relate to the content, causing some loss of information.

We propose Ksformer, which is made up of MKRA and LFPM. MKRA estimates queries in windows of different sizes and then uses a top-k operator to select the most important k queries. This approach enhances computational efficiency and incorporates content-aware capabilities. Meanwhile, multi-scale windows adeptly manage blurs of varying sizes. LFPM employs lightweight parameters to extract spectral features. The contributions of this work are summarized as:

- Ksformer extracts spectral features with ultra-lightweight parameters, performing MKRA in both spatial and frequency domains and then fusing them, which narrows the gap between clean and hazy images in terms of both space and spectrum.
- Ksformer achieves a PSNR of 39.4 and an SSIM of 0.998 with only 5.8M parameters, which is significantly better than other state-of-the-art methods.

2. Method

2.1 Image Dehazing

We use three encoders and three decoders and downsample by 4 × 4 for a compact model. We use Multi-scale Key-select Routing Attention Module (MKRAM) only in the smaller dimensions to reduce computational complexity. For any given input feature map \( X \in R^{H \times W \times C} \), first, we divide it into four parts along the channel dimension, with window sizes of 2 × 2, 4 × 4, 8 × 8, 64 × 64. Then, it is divided into \( S \times S \) non-overlapping regions. Each region contains \( \frac{HW}{S^2} \) feature vectors. After this step, \( X \) is reshaped into \( X' \in R^{2^4 \times \frac{HW}{S^2} \times C} \). Then, we use linear projection to weight and derive \( Q, K, V \in R^{S^2 \times \frac{HW}{S^2} \times C} \).

\[
Q = X'W^q, K = X'W^k, V = X'W^v, \tag{1}
\]

Here, \( W^q, W^k, W^v \in R^{C \times C} \) represent the weights for \( Q, K, \) and \( V \), respectively. We construct an Attention module to identify the areas where important key-value pairs are located. In simple terms, we use the average values of each region to derive region-level queries and keys, \( Q_r, K_r \in R^{r^2 \times \frac{C}{2}} \). Then, we derive the region-to-region importance association matrix using the following formula.

\[
A^{r \times n}_{r} = \text{Softmax} \left( \frac{Q_r(K_r)^T}{\sqrt{C}} \right), \tag{2}
\]
Modulate, Conv 3 × 3

Conv 3 × 3

Fig. 1 The architecture of the proposed Ksformer.

Input

Conv 1 × 1

Conv 3 × 3

Global Branch

Local Branch

Conv 1 × 1

MKRAM

Fig. 2 (a) is the architecture of the proposed MKRAM. (b) is the architecture of the proposed LFPM.

Here, \( A^{s_r \times s_t}_{k} \) represents the degree of association between two regions. \( n \times n \) represents the size of the window. Next, we concatenate \( A^{s_r \times s_t}_{k} \) along the channel dimension to obtain \( A_{r} \in R^{S^2 \times C} \). Next, we retain the top k most important queries using the top-k operator and prune the association graph to derive the index matrix.

\[
I_r = \text{topk}(A_r),
\]

Here, \( I_r \in R^{S^2 \times K} \). So, the i-th row of \( I \) contains the k indices of the most relevant regions for the i-th region. Using the importance index matrix \( I_r \), we can capture long-range dependencies, be content-aware, and reduce computational complexity. For each query and token in region \( i \), it will focus on all key-value pairs in the union of \( k \) important regions indexed by \( I^{(i,1)}, I^{(i,2)}, \ldots, I^{(i,k)} \). We collect the key and value tensors.

\[
K^g = \text{gather}(K, I_r), V^g = \text{gather}(V_r),
\]

We collect the key and value tensors. Here \( K^g, V^g \in R^{S^2 \times \frac{1}{2}S^2 \times C} \). Finally, we focus our attention on the collected key-value pairs.

Output = Attention \((Q, K^g, V^g)\). (5)

3. Lightweight Frequency Processing Module

Past approaches often employed wavelet [29]–[32] or Fourier [33] transformations to divide image features into multiple frequency sub-bands. These approaches raised the computational load for reverse transformation and didn’t boost key frequency elements. To solve this, we added a lightweight module for spectral feature extraction and modulation. It efficiently splits the spectrum into different frequencies and uses a small number of learnable parameters to emphasize the most informative ones.

4. Multi-scale Key-select Routing Attention Module

As shown in Fig.2, to improve model efficiency, our MKRAM module processes the spatial domain, low frequencies, and high frequencies in parallel and then fuses the three outputs.

Table 1 Quantitative comparisons with SOTA methods on the RESIDE-Indoor [34] and Haze4K [27] datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>RESIDE-Indoor [34]</th>
<th>Haze4K [27]</th>
<th># Params</th>
<th># GFLOPs</th>
</tr>
</thead>
<tbody>
<tr>
<td>(ICCV’17) AOD-Net [3]</td>
<td>35.82 ( \pm ) 0.684</td>
<td>17.15 ( \pm ) 0.83</td>
<td>0.06M</td>
<td>-</td>
</tr>
<tr>
<td>(ICCV’19) GridDehazeNet [7]</td>
<td>32.16 ( \pm ) 0.984</td>
<td>22.99 ( \pm ) 0.85</td>
<td>3.15M</td>
<td>228.34</td>
</tr>
<tr>
<td>(AAAI’20) FFA-Net [4]</td>
<td>36.39 ( \pm ) 0.989</td>
<td>26.96 ( \pm ) 0.95</td>
<td>4.88M</td>
<td>208.34</td>
</tr>
<tr>
<td>(CVPR’20) MSBDN [9]</td>
<td>33.79 ( \pm ) 0.984</td>
<td>22.09 ( \pm ) 0.85</td>
<td>3.15M</td>
<td>41.58</td>
</tr>
<tr>
<td>(CVPR’19) KDNN [13]</td>
<td>34.72 ( \pm ) 0.985</td>
<td>-</td>
<td>5.99M</td>
<td>-</td>
</tr>
<tr>
<td>(CVPR’21) ALICNet [5]</td>
<td>37.17 ( \pm ) 0.990</td>
<td>-</td>
<td>2.63M</td>
<td>26.10</td>
</tr>
<tr>
<td>(CVPR’22) DehazeNet [14]</td>
<td>36.63 ( \pm ) 0.984</td>
<td>-</td>
<td>2.63M</td>
<td>59.14</td>
</tr>
<tr>
<td>(ICCV’21) KEDNet [10]</td>
<td>36.41 ( \pm ) 0.990</td>
<td>33.49 ( \pm ) 0.98</td>
<td>18.9M</td>
<td>-</td>
</tr>
</tbody>
</table>

Output = Attention \((Q, K^q, V^q)\).
5. Experiments

5.1 Implementation Details

During our experiments, we employ PyTorch version 1.11.0 and utilize the capabilities of four NVIDIA RTX 4090 GPUs to perform all tests. In the training phase, images are randomly cropped into 320 × 320 pixel patches. For assessing the model’s computational complexity, we adopt a size of 128 × 128 pixels. The Adam optimizer is engaged for optimization, with decay rates set at 0.9 for $\beta_1$ and 0.999 for $\beta_2$. The initial learning rate is configured at 0.00015, and we apply a cosine annealing strategy for its scheduling. The batch size is maintained at 64. Through empirical determination, we set the penalty parameter $\lambda$ to 0.2 and $\gamma$ to 0.25, and we proceed with training for 80,000 iterations.

5.2 Quantitative and Qualitative Experiments

Visual Comparison. To thoroughly assess our method, we tested it on both the synthetic Haze4K [27] dataset and the real-world RTTS [34] dataset. As shown in Fig.3 and Fig.4, it’s clear that our method outperforms others in terms of edge sharpness, color fidelity, clarity of texture details, and handling of sky areas, whether on synthetic or real datasets. Quantitative Comparison. We quantitatively compared Ksformer with the current state-of-the-art methods on the SOTS indoor [34] and Haze4K [27] datasets. As shown in Table.1, for the SOTS indoor [34] dataset, Ksformer achieved a PSNR of 39.40 and an SSIM of 0.994, which is a 0.09 PSNR improvement over the second-best method, and it did so with only 30% of the parameter volume. For the Haze4K [27] dataset, Ksformer reached a PSNR of 33.74 and an SSIM of 0.98. The quantitative comparison fully demonstrates that Ksformer outperforms other state-of-the-art methods in terms of performance.

5.3 Ablation Study

To prove the effectiveness of our method, we conducted an ablation study. We first built a U-Net as the base network, and then gradually added modules to the baseline. As shown in Table.2, both PSNR and SSIM improved with the step-by-step addition of modules, and the metrics reached their best values after effectively combining the modules we proposed.

6. Conclusion

This paper introduces Ksformer, which combines a top-k operator with multi-scale windows, giving the network the char-
acteristics of content awareness and low complexity. At the same time, it obtains spectral features with ultra-lightweight parameters, narrowing the spectral gap between clean and foggy images. On the SOTS indoor [34] dataset, it achieved a PSNR of 39.4 and an SSIM of 0.994 with only 5.8M.

Although the Ksformer has a relatively small parameter count of just 5.8 million, it’s unfortunate that it can’t be implemented on embedded systems due to its high GFLOPs. We plan to further explore the balance between performance and computational complexity. By appropriately reducing the number of channels and modules, we aim to make the Ksformer suitable for embedded systems, allowing it to play a significant role in a broader range of fields.

Acknowledgments

This work was supported in part by the Youth Science and Technology Innovation Program of Xiamen Ocean and Fisheries Development Special Funds (23ZHJB0390CB24), Xiamen Ocean and Fisheries Development Special Funds (22CB013H04).

References

