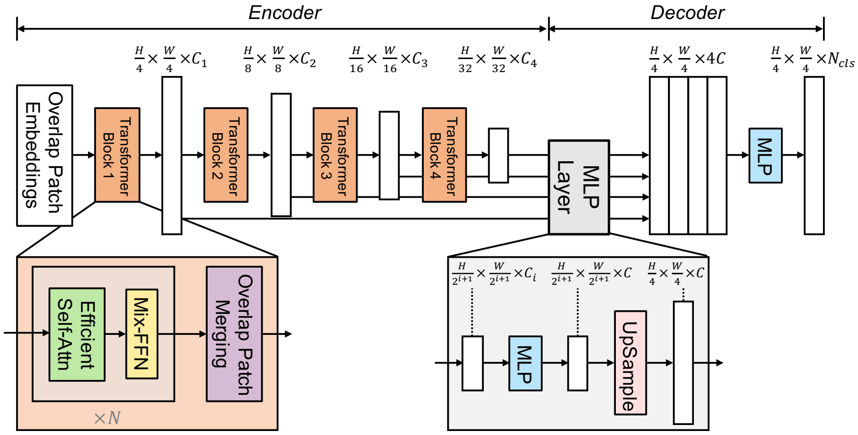
Encoder
Decoder
$\frac{H}{4}\times\frac{W}{4}\times C_1$
$\frac{H}{8}\times\frac{W}{8}\times C_2$
$\frac{H}{16}\times\frac{W}{16}\times C_3$
$\frac{H}{32}\times\frac{W}{32}\times C_4$
$\frac{H}{4}\times\frac{W}{4}\times 4C$
$\frac{H}{4}\times\frac{W}{4}\times N_{cls}$
Overlap Patch Embeddings
Transformer Block 1
Transformer Block 2
Transformer Block 3
Transformer Block 4
MLP Layer
MLP
Efficient Self-Attn
Mix-FFN
Overlap Patch Merging
$\times N$
$\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}\times C_i$
$\frac{H}{2^{i+1}}\times\frac{W}{2^{i+1}}\times C$
$\frac{H}{4}\times\frac{W}{4}\times C$
MLP
UpSample

Figure 2: **The proposed SegFormer framework** consists of two main modules: A hierarchical Transformer encoder to extract coarse and fine features; and a lightweight All-MLP decoder to directly fuse these multi-level features and predict the semantic segmentation mask. "FFN" indicates feed-forward network.

29]; introducing boundary information [30–37]; designing various attention modules [38–46]; or using AutoML technologies [47–51]. These methods significantly improve semantic segmentation performance at the expense of introducing many empirical modules, making the resulting framework computationally demanding and complicated. More recent methods have proved the effectiveness of Transformer-based architectures for semantic segmentation [7, 46]. However, these methods are still computationally demanding.

Transformer backbones. ViT [6] is the first work to prove that a pure Transformer can achieve state-of-the-art performance in image classification. ViT treats each image as a sequence of tokens and then feeds them to multiple Transformer layers to make the classification. Subsequently, DeiT [52] further explores a data-efficient training strategy and a distillation approach for ViT. More recent methods such as T2T ViT [53], CPVT [54], TNT [55], CrossViT [56] and LocalViT [57] introduce tailored changes to ViT to further improve image classification performance.

Beyond classification, PVT [8] is the first work to introduce a pyramid structure in Transformer, demonstrating the potential of a pure Transformer backbone compared to CNN counterparts in dense prediction tasks. After that, methods such as Swin [9], CvT [58], CoaT [59], LeViT [60] and Twins [10] enhance the local continuity of features and remove fixed size position embedding to improve the performance of Transformers in dense prediction tasks.

Transformers for specific tasks. DETR [52] is the first work using Transformers to build an end-to-end object detection framework without non-maximum suppression (NMS). Other works have also used Transformers in a variety of tasks such as tracking [61, 62], super-resolution [63], ReID [64], Colorization [65], Retrieval [66] and multi-modal learning [67, 68]. For semantic segmentation, SETR [7] adopts ViT [6] as a backbone to extract features, achieving impressive performance. However, these Transformer-based methods have very low efficiency and, thus, difficult to deploy in real-time applications.

3 Method

This section introduces SegFormer, our efficient, robust, and powerful segmentation framework without hand-crafted and computationally demanding modules. As depicted in Figure 2, SegFormer consists of two main modules: (1) a hierarchical Transformer encoder to generate high-resolution coarse features and low-resolution fine features; and (2) a lightweight All-MLP decoder to fuse these multi-level features to produce the final semantic segmentation mask.

Given an image of size $H \times W \times 3$, we first divide it into patches of size 4×4. Contrary to ViT that uses patches of size 16×16, using smaller patches favors the dense prediction task. We then use these patches as input to the hierarchical Transformer encoder to obtain multi-level features at {1/4, 1/8, 1/16, 1/32} of the original image resolution. We then pass these multi-level features to the All-MLP decoder to predict the segmentation mask at a $\frac{H}{4} \times \frac{W}{4} \times N_{cls}$ resolution, where N_{cls} is the