Depth estimation with Visual Transformer

Lee Ju Hwan, Dang Thanh Vu, Yu Gwang Hyun and Kim Jin Young

Abstract: In this study, we propose a novel approach to estimate depth information in a single image. Our built-in model utilizes Visual Transformer (ViT), a novel architecture with great success in solving a wide range of computer vision problems. The self-attention mechanism enables a model to capture the correlation between multi patches of images which takes depth information into considerations across a wide image area. This ability is beneficial to estimate depth information in a single image since global awareness from self-attention is superior to local awareness from convolution-based models. Our proposed framework is simple yet effective. The quantitative results show that the self-attention-based encoder outperforms the convolution-based encoder in terms of varied metrics. Besides, the qualitative results also illustrate predictive heatmap of depth is equitable.

Introduction

Method

Network Architecture

This study introduces an effective yet simple framework for depth estimation in a single image. Our framework constitutes of an encoder and a decoder. The encoder is deployed with the Visual Transformer (ViT) model, where self-attention is implemented to capture correlations among patches of the input image. The decoder is simply consecutive upsampling layers with convolution to reconstruct a depth map. Our framework is illustrated in Figure 1. The proposed network can be trained end-to-end with a regression-objective function such as loss.

Encoder

We manipulate the Visual Transformer model to extract a visual representation of given images. A main component of the ViT model is the self-attention encoder module, which implements Transformer architecture in the most standard way. According to the original version of ViT, our encoder involves three main steps.

Patch embedding: we split an image into a sequence of patches and map each patch to a D dimensions embedding space. Precisely, we put an image x∈R^(C×W×H) through D 2d convolutions with the kernel size of P×P and stride of P, resulting in a feature map with the size D×N×N, then flattening the feature map into a sequence of N^2 latent vectors with a constant size of D. In the above configuration, N^2=H×W/P^2 is the number of embedded patches, where (H,W) represesents the resolution of the original image, and (P,P) is the resolution of an image patch. Apart from the embedded patches, the ViT model adds an extra learnable class embedding for classification tasks, and the results obtained from using this class token are referred to as "ViT-class" in our study. To maintain the position of the patches after flattening, we follow the standard way by adding a learnable 1D positional embedding into each patch, and this positional embedding does not share weights across patches.

Multi-head self-attention:

* The selfattention mechanism is an integral component of Transformers, which explicitly models the interactions between all entities of a sequence for structured prediction task.
* The main difference of self-attention with convolution operation is that the weights are dynamically calculated instead of static weights (that stay the same for any input) as in the case of convolution.
* The goal of self-attention is to capture the interaction amongst all n entities by encoding each entity in terms of the global contextual information

Encoder is a computational block consisting of a multi-head attention module [10] and an MLP with two consecutive linear layers. Input and output of encoder module are both embedded vector of batches. LayerNorm is applied before feeding embedded vectors into the attention module and MLP. The multi-head attention module expands the model's ability to jointly focus on different positions, thus providing different representation subspaces of pair (key K,query Q,value V) from different attention heads:

multihead=Concat(head\_1,…,head\_h ),

(1)

Where each head is a context vector from scale dot-product attention.

head\_i=softmax((Q\_i K\_i^T)/√d) V\_i, (2)

Q, K, V represents query, key and value, respectively calculated inside the transformer architecture that encodes information from the image's patches with the self-attention mechanism to mutually attend to each other. d is the dimensions of patch embeddings, which equation (2) employs d to scale the attention scores.

Decoder:

Although the key idea of this study is to implement self-attention mechanism integrated in decoder for extract depth information from a single image, we also pay attention to modify the decoder part. As shown in the experimental results section, a plain decoder with upsampling convolutional layers show some defective when reconstructing the depth maps. The reason is that patch’s embeddings from ViT model lacks of local textures. The self-attention mechanism can well capture global information through correlation of patches of images, but the local feature such as edges and fine textures are omitted [cite]. A direct answer to this problem is training ViT model with a huge dataset for a long run, but it is impractical for transfer learnning. Therefore, we present in this paper the MultiScaling decoder, a modified version of decoders that have been use extensively in U-Net architectures. Our MultiScaling decoder derives multi resolutions feature maps of the original input and concatenates each feature map at different resolution with output of correseponding decoder layer.

For details, the output of encoder layer is array of patch’s embeddings , when the size of image is and the patch size is , and . We first arrange the collection of patch embeddings into a tensor, , by simply reshape the output of the encoder. The emebding tensor is input of the decoder module to reconstruct depth map with the same size of the original image. To constitute the depth map, we employ 4 consectuative upsampling layers, each layer output size is double of the previous layer. To integrate visual information to the decoder, we simultaneously dowmsample the original image into 4 corresponding resolution level and concatenate it with each output of decoder’s layer. The upsamling layer is sequence of 2 convolutional layers followed by a leaklyRelu activation function. While the dowmsampling layer is a convolutional layer with the stride step is same with the kernel size and without padding. Figure 1 illustrate our framework with ViT encoder and MultiScaling decoder, the size of each layer is also mentioned in the figure.

Experimental results

1. Dataset and preprocessing
2. Quantitative results
3. Qualitative results

Conclusions

Acknowledgments

References