Project Report - ECE 285

Logan Reuter
MAE
A69035755

Abstract

Pose estimation has grown in popularity with the rise of AR/VR technologies, allowing for digital input without a physical interface. A new approach to this technology has been proposed with the ViTPose [1] model that utilizes a plain and non-hierarchical vision transformer to accomplish the pose estimation. In this project, we re-implement this model using a new dataset to validate the feasibility of this model for use in future products.

1. Introduction

In recent years, there has been a growth in popularity in AR/VR technology with devices like the Apple Vision Pro and Meta Quest 2. These devices use the user's hand gestures to register inputs, which is accomplished by using pose estimation. This method uses a neural network to locate several key locations on a body (e.g. hands, elbows, knees, etc.) and attaches points to these locations. This technology can also be adapted to be used with robotics, allowing for remote-less communication with autonomous robots. What makes this task so challenging is that numerous occlusions can make it difficult to locate body parts; these occlusions can occur due to wearing certain types of clothes (e.g. gloves) or by being obstructed by objects. Appearance can also vary greatly from person to person, which adds another layer of difficulty. Addressing these issues can greatly increase the feasibility of this technology in future products.

2. Related Works

Human keypoint detection is a very popular area of research within computer vision due to its wide number of uses, and as such a large number of implementations exists. Previous research into keypoint detection, like Mask R-CNN [2], utilize CNN networks. Later transformers were adopted for use as the decoder due to their superior performance but they still utilized CNNs to extract features, e.g. TransPose [3]. The method proposed by ViTPose [1] utilizes a plain vision transformer (ViT) for the entirety of the computation, using the model proposed in [4] as the backbone for the transformer. This approach has shown promising results, achieving state-of-the-art performance on the MS COCO Keypoint dataset.

3. Methods

The method used in this paper follows the method described in [1], which implements a plain vision transformer based on the ViT model described in [4]. The only prepocessing performed

in this paper is all images are resized to (256×256) before being passed to the patch embedder. The patch embedder receives an image of size $x_{\text{in}} \in \mathbb{R}^{256\times256\times3}$ and converts it into a flattened sequence of (16×16) patches, resulting in an output $x_{\text{out}} \in \mathbb{R}^{256\times768}$. Each patch is embedded with positional data to be used later. After the patch embeddeding the data enters the backbone of the model, which is 12 transformer blocks. These blocks consist of a Multi-Head Self Attention block and a Feed Forward Network block. Before each block a layer normalization is applied, with a residual connection after each block (See Figure 1). The Feed Forward Network is comprised of 2 linear layers and a GELU non-linearity.

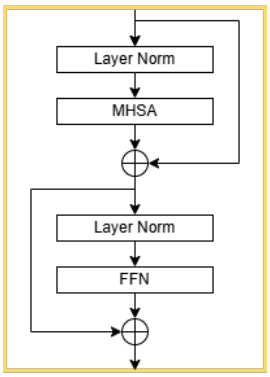

Figure 1: Diagram of the transformer block

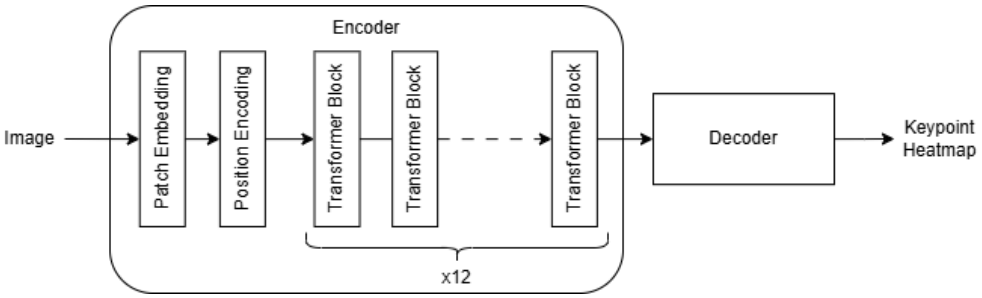

Figure 2: Full Vision Transformer network

The final part of the model is a decoder block which is two deconvolution layers, with a kernel of 4 and a stride of 2, followed by a (1×1) convolutional layer which is used to get the localization heatmaps for the keypoints (see Figure 3). The final output is $x_{\text{final}} \in \mathbb{R}^{64\times64\times N_k}$, where N_k is the number of keypoints in the dataset.

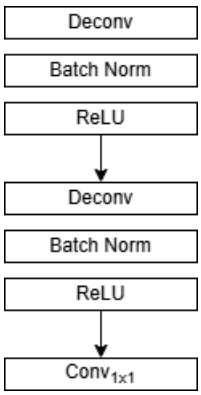

Figure 3: Classic Decoder

For training, an AdamW optimizer is used with a learning rate of $5e$-4 and a weight decay of 0.1. A weighted MSE loss is used to evaluate the loss of the model during training where the weight is a visibility vector (1 is visible and 0 is not visible) provided by the dataset. The model was trained for 210 epochs on 1000 images. Traditionally, the percentage of correct keypoints is used to evaluate the correctness of the Leeds Sports Pose dataset, however, the AP_{50} metric was selected to align with the metrics used by [1]. AP_{50} works by first determining the Object Keypoint Similarity (OKS) (Equation 1) as described by the MS-COCO keypoint evaluation metrics [5]. Here d_i is the Euclidean distance of between the keypoint and ground truth, s is the scale of the ground truth object, k_i is $2 \times \sigma$ which are the per-keypoint standard deviation. The $\delta(v_i > 0)$ is the Dirac-Delta function used to find the number of keypoints with a non-zero visibility.

$$\text{OKS} = \frac{\Sigma_i \exp\left(-\frac{d_i}{2s^2k_i^2}\right)\delta(v_i > 0)}{\Sigma_i \delta(v_i > 0)} \tag{1}$$

4. Experiments

The dataset choosen for this evaluation was the Leeds Sports Pose extended (LSPe) dataset [6], which consists of 10,000 images collected from Flickr using searches for 'parkour', 'gymnastics', and 'athletics'. The data is stored as JPEGs with an additional .mat file which contains the annotated keypoints for each image. This dataset contains only 14 keypoints (ankles, knees, hips, wrists, elbows, shoulders, neck, and top of the head) as compared to MS-COCO which uses 17. Despite containing 10000 images, a subset of only 2000 images was used in the training and evaluation of the model due to training constraints.

Before training the model on the full dataset, a sanity check was performed using the first 4 images of the dataset. The model would be repeatedly (5500 iterations) trained on the same four images until the loss was driven to near 0. The purpose of this sanity check was to ensure all components of the model worked properly and to gain insight into the outputs of the model, like the heatmaps (see Figure 4).

Figure 4: Heatmaps produced for right ankle keypoint after 2800 iterations

With the sanity check proving the functionality of the model, the model was then trained on the actual dataset. After 210 epochs, the model received an AP_{50} score of 96 (see Figure 5). For the testing and validation training sets, the model received similar scores of 97 and 96 respectively. These scores are are similar to the scores found by the original ViTPose model [1] score of 90.7 for the base model.

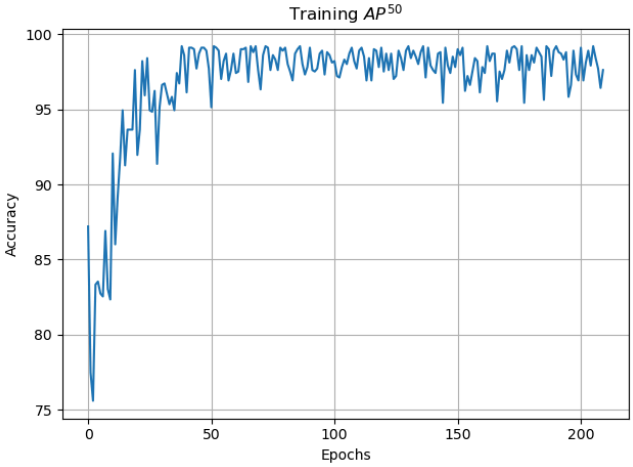

Figure 5: AP_{50} scores during training

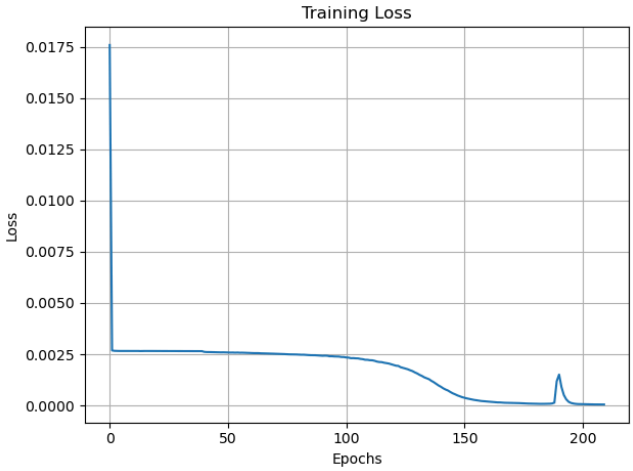

Figure 6: MSE Loss during training

While these scores are a good way to validate a model on a dataset, it is difficult to compare across datasets without standardized parameters because the σ and s_i parameters can differ between datasets. The scale parameter for MS-COCO is provided by the dataset and measures the area of the bounding box, however, Leeds Sports Pose did not provide such value. Instead, the torso length was used as the scale value for the purposes of this paper. The σ values provide a standard deviation for each keypoint and are fine-tuned to a dataset, these values are predetermined for the MS-COCO set. With some preliminary testing, values were found for use with the Leeds Sports Pose (see Table 1) but a more comprehensive evaluation would be required to confirm their correctness. These values play a large role in determining the OKS scores which makes using accurate values crucial.

Keypoints	σ_i
Ankles	4.45
Knees	4.35
Hips	5.35
Wrists	3.1
Elbows	3.6
Shoulders	3.95
Neck	2.5
Top of Head	1.3

Table 1: σ values used in OKS calculations

The results found by this study confirm the findings of the original paper and shows that a simple vision transformer can serve as a solid option for pose estimation tasks. The model is able to accurately predict keypoints as seen by Figure 7.

Figure 7: Visual pose estimation results on some test images (red are predicted keypoints and blue are ground truth)

5. References

[1] Y. Xu, J. Zhang, Q. Zhang, and D. Tao, "ViTPose: Simple Vision Transformer Baselines for Human Pose Estimation," 2022, [Online]. Available: https://arxiv.org/abs/2204.12484

[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," 2018, [Online]. Available: https://arxiv.org/abs/1703.06870

[3] S. Yang, Z. Quan, M. Nie, and W. Yang, "TransPose: Keypoint Localization via Transformer." [Online]. Available: https://arxiv.org/abs/2012.14214

[4] A. Dosovitskiy *et al.*, "An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale," 2021, [Online]. Available: https://arxiv.org/abs/2010.11929

[5] M. COCO, "Keypoint Evaluation." [Online]. Available: https://cocodataset.org/#keypoints-eval

[6] S. Johnson and M. Everingham, "Learning Effective Human Pose Estimation from Inaccurate Annotation," in *Proceedings of Computer Vision and Pattern Recognition (CVPR) 2011*, 2011.

6. Appendix

6.1. Implementation

A majority of the code for this project was implemented from scratch, however, a few pieces were reused or inspired by external sources. The implementation for the OKS function was borrowed from an article by learnopencv (link). Various components for training and checking accuracy are based on the code provided in the previous homeworks of the course.