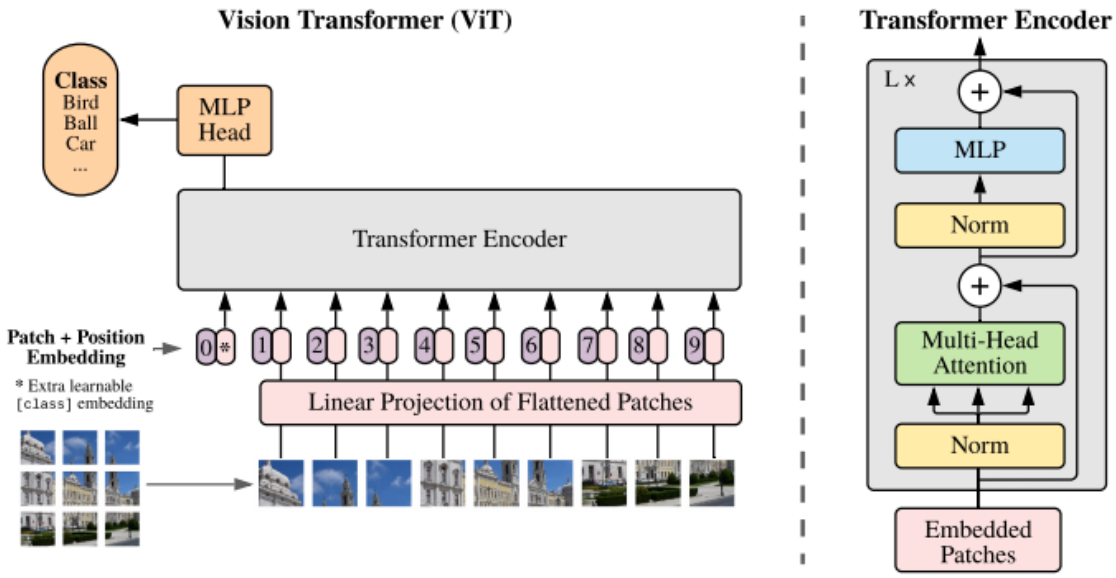
AN IMAGE IS WORTH 16X16 WORDS:
TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

ABSTRACT

Transformer Architecture가 NLP tasks에서 사실상의 표준이 되었지만, Computer Vision 분야에의 적용은 아직 제한적이다.
우리는 CNN에 의존하지 않고도 Transformer만을 사용해 image classification task를 수행할 수 있음을 보여주려고 한다. 이미지를 image patches의 sequence로 처리하고, 대량의 데이터에 대해 pre-trained한 후 작은 image recognition benchmarks(ImageNet, CIFAR-100) 에 적용시킨다.
그 결과 Vision Transformer (ViT) 는 다른 모델과 비교해서 좋은 성능을 얻을 수 있었고, 동시에 computational resources를 현저히 적게 사용하였다.

MODEL



이미지를 고정된 크기의 patch로 쪼개고 각각 linear하게 embedding한 후 position embedding을 더해 결과 벡터를 transformer encoder의 입력으로 input한다.
Classification task를 수행하기 위해 classification token을 만들어서 sequence에 추가한다.

METHOD

1. Vision Transformer (ViT)

일반적인 Transformer는 token embedding에 대한 1차원 sequence를 입력으로 받는다. 따라서 2차원의 이미지를 다루기 위해 이미지를 flattened 2차원 patch sequence로 reshape한다.
즉 H * W * C를 N * (P^2 * C) 로 변환한다. H*W는 원본 이미지의 크기, C는 채널의 개수, (P,P)는 image patch의 크기이고 N = HW / P^2 은 patch의 개수이다.
Transformer는 모든 레이어에서 고정된 벡터 size D를 사용하기 때문에 이미지 패치를 flatten한 뒤

D차원의 벡터로 linear projection 시킨다.
또한 BERT의 [CLASS] 토큰과 비슷하게 Embedding된 patch의 sequence에 z0 = x_class embedding을 추가로 붙여넣는다.
이후 나온 encoder output은 이미지 representation으로 해석해서 classification에 사용한다.

$$\mathbf{z}_0 = [\mathbf{x}_{\text{class}}; \mathbf{x}_p^1\mathbf{E}; \mathbf{x}_p^2\mathbf{E}; \cdots; \mathbf{x}_p^N\mathbf{E}] + \mathbf{E}_{pos}, \qquad \mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}, \ \mathbf{E}_{pos} \in \mathbb{R}^{(N+1) \times D} \tag{1}$$

$$\mathbf{z}'_\ell = \text{MSA}(\text{LN}(\mathbf{z}_{\ell-1})) + \mathbf{z}_{\ell-1}, \qquad \ell = 1 \ldots L \tag{2}$$

$$\mathbf{z}_\ell = \text{MLP}(\text{LN}(\mathbf{z}'_\ell)) + \mathbf{z}'_\ell, \qquad \ell = 1 \ldots L \tag{3}$$

$$\mathbf{y} = \text{LN}(\mathbf{z}_L^0) \tag{4}$$

이는 위치 정보를 활용할 수 있도록 한 위치 embedding이다. 학습 가능한 1차원의 embedding을 사용하는데, 2차원 정보를 유지하는 위치 embedding도 활용해 보았으나 유의미한 성능 향상은 없었기 때문이다.

2. FINE-TUNING AND HIGHER RESOLUTION

ViT는 large dataset에 대해 pre-train한 후 더 작은 downstream task에 fine-tune하는 방법을 사용한다. Fine-tuning 시에는 pre-train된 prediction head를 제거하고 0으로 초기화된 D*K feedforward layer를 연결한다. 이때 K는 downstream class의 개수이다.
Fine-tuning 단계에서는 더 높은 해상도에서 학습하는 것이 정확도 향상에 효과적이라는 것이 알려져 있다. 따라서 더 높은 해상도의 이미지를 처리해야 할 경우 patch의 크기를 같게 유지한다.
이러한 해상도 조절과 patch 추출 방식은 이미지의 2차원 구조에 대한 inductive bias를 다루는 유일한 point이다.

3. EXPERIMENTS

Model	Layers	Hidden size D	MLP size	Heads	Params
ViT-Base	12	768	3072	12	86M
ViT-Large	24	1024	4096	16	307M
ViT-Huge	32	1280	5120	16	632M

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21K (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

Table 2: Comparison with state of the art on popular image classification benchmarks. We report mean and standard deviation of the accuracies, averaged over three fine-tuning runs. Vision Transformer models pre-trained on the JFT-300M dataset outperform ResNet-based baselines on all datasets, while taking substantially less computational resources to pre-train. ViT pre-trained on the smaller public ImageNet-21k dataset performs well too. *Slightly improved 88.5% result reported in Touvron et al. (2020).

3D human pose estimation in video with temporal convolutions and semi-supervised training

ABSTRACT

이 논문에서는 비디오 환경에서 2차원 keypoint들에 대해 diated temporal convolution 기반의 fully convolutional model을 적용해 3차원 pose estimation을 수행한다.
또한 unlabeled video data를 활용하는 간단하고 효율적인 semi-supervised 학습 방법인 back-projection을 소개한다.
Supervised setting에서 이 논문은 Human3..6M 데이터에서 11% 정도의 error reduction을 달성하였다.

MODEL

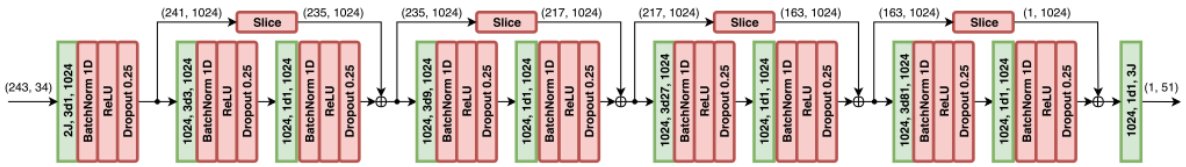


Figure 2: An instantiation of our fully-convolutional 3D pose estimation architecture. The input consists of 2D keypoints for a recpetive field of 243 frames ($B = 4$ blocks) with $J = 17$ joints. Convolutional layers are in green where `2J, 3d1, 1024` denotes $2 \cdot J$ input channels, kernels of size 3 with dilation 1, and 1024 output channels. We also show tensor sizes in parentheses for a sample 1-frame prediction, where `(243, 34)` denotes 243 frames and 34 channels. Due to valid convolutions, we slice the residuals (left and right, symmetrically) to match the shape of subsequent tensors.

이 모델은 2차원 keypoint sequences를 입력으로 받아 temporal convolution을 통해 변환하는 residual connection이 포함된 fully convolutional 구조이다. CNN 모델은 parallelization을 가능하게 하지만 RNN은 시간에 따른 parallelization을 할 수 없다. Convolution Model에서 path of the gradient는 고정된 길이를 가지기 때문에, Vanishing 문제 및 exploding gradient 문제를 완화시킬 수 있다.
입력 layer는 각 joint J의 연결된 (x,y) 좌표를 취하게 되고, 커널 크기인 W 및 출력 채널인 C와 함께 temporal convolution을 적용한다. 그 다음 skip-connection으로 연결된 B(ResNet-style black)이 이어진다. 각 블록은 먼저 W 및 dilation factor D = W^B로 1D conv를 수행한 다음 커널 크기 1로 convolution을 수행하게 된다. Convolution 이후에는 Batch Normalization, ReLU, Dropout이 이어진다. 마지막 layer는 시간 정보를 활용하기 위해 과거 및 미래의 모든 데이터를 사용해 sequence의 모든 frame에 대한 pose estimation을 출력한다.

Semi-Supervised approach

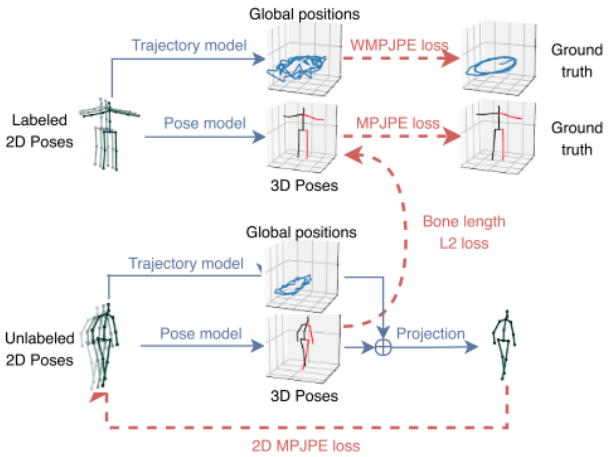


Figure 3: Semi-supervised training with a 3D pose model that takes a sequence of possibly predicted 2D poses as input. We regress the 3D trajectory of the person and add a soft-constraint to match the mean bone lengths of the unlabeled predictions to the labeled ones. Everything is trained jointly. WMPJPE stands for "Weighted MPJPE".

이 논문에서는 공개되어 있는 video를 2차원 keypoint 탐지기와 결합하여 back-projection loss term으로 supervised loss function을 확장한다. 따라서 unlabeled data에 대한 auto-encoding 문제를 해결하고 encoder는 pose estimator, decoder는 projection layer의 역할을 수행한다.
따라서 decoder의 2차원 joint 좌표가 원래 입력에서 멀리 떨어져 있는 경우 페널티를 부여받는다. 총 두 가지 목표가 있는데, labeled data가 batch의 first-half를 차지하고 unlabeled data가 batch의 나머지를 차지하도록 최적화 시킨다. Labeled data의 경우 GT 3차원 pose를 사용하고 supervised loss를 학습한다. Unlabeled data는 예측된 3차원 pose가 2차원으로 다시 투영된 다음 autoencoder의 Loss를 구현하는 데 사용한다.

RESULT

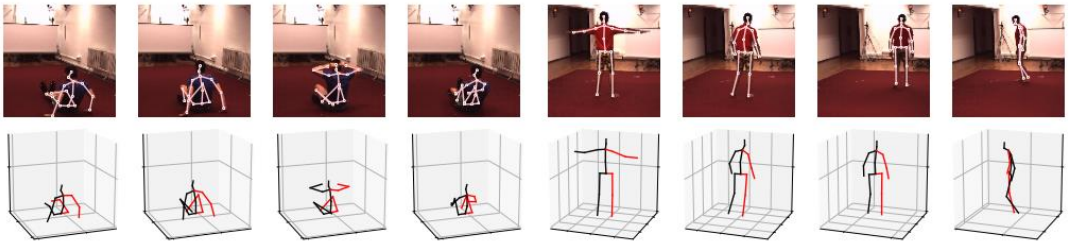


Figure 4: Qualitative results for two videos. **Top:** video frames with 2D pose overlay. **Bottom:** 3D reconstruction.

	Dir.	Disc.	Eat	Greet	Phone	Photo	Pose	Purch.	Sit	SitD.	Smoke	Wait	WalkD.	Walk	WalkT.	Avg
Single-frame	12.8	12.6	10.3	14.2	10.2	11.3	11.8	11.3	8.2	10.2	10.3	11.3	13.1	13.4	12.9	11.6
Temporal	3.0	3.1	2.2	3.4	2.3	2.7	2.7	3.1	2.1	2.9	2.3	2.4	3.7	3.1	2.8	2.8

Table 2: Velocity error over the 3D poses generated by a convolutional model that considers time and a single-frame baseline.