

Comparison of Various CNN Models for 3D Human Pose Estimation

Hyunjin Bae

I. INTRODUCTION

3D human pose estimation을 더 효율적이고 더 좋은 성능을 내기 위해 다양한 시도들과 연구들이 지금까지 있었다. 그 중에서도, 본 인턴 기간 동안은 HMR [1] 을 기반으로하여 parameter 수를 줄이고, model을 가볍게 만들면서 성능을 유지하는 것을 목표로 하고 있다. 경량화를 진행하기에 앞서서, HMR [1] 은 Encoder 부분에서 ResNet50을 backbone으로 사용하고 있다. 논문이 2018년도에 나왔고, ResNet50이 2015년도에 나온 것을 고려하면, 다른 여러 CNN model 들 중에서도 ResNet50을 backbone으로 한 이유가 있을지 궁금해진다. 그 사이에 여러 성능 좋은 많은 모델들 [2], [3], [4]이 나왔기 때문에 그에 비하면 ResNet50은 많이 무거운 model이다.

따라서 ResNet50 이후에 나온 여러 model들에 대한 공부와 함께 HMR의 encoder부분의 ResNet50을 다른 여러 model 들로 대체했을 때의 train 및 evaluation 결과를 비교해보고자 한다. 그래서 각 모델의 parameter 수와 성능을 비교했을 때 가장 가벼우면서 성능이 높은 model을 backbone으로 삼아, 다양한 model compression 기법을 사용해서 경량화된 3D Human pose estimation model을 만들고자 한다.

원래, HMR [1]에서는 adversarial prior로써 discriminator를 함께 학습시킨다. 논문에서는 reprojection loss가 2D joint의 위치를 설명할 수 있는 3D body를 만들어내지만 인체학적으로 너무 불가능한 body에 대해서는 여전히 잘 학습을 못하는데 이 문제를 조금 regularize하기 위해 discriminator을 사용하였다고 주장한다 [1]. 하지만 직관적으로 생각했을 때 Discriminator를 하나 추가하는 것이 정말 더 학습이 잘 되도록 해주는데 얼마나 큰 영향을 미치는지에 대해서 감이 잡히지 않았다. 그런다, SPIN [5]을 보면서 Discriminator 없이도 학습이 잘 될 수 있지 않을까라는 생각을 했다. 따라서 우선은 HMR의 Encoder 부분에만 집중해서 여러가지 CNN 모델을 적용시켜보고 이후 각각의 모델에 prior를 추가했을 때, 효과를 보고자한다.

II. VARIOUS CNN MODEL

비교해보자 하는 CNN model 들에는 DenseNet [2], Xception [4], MobileNet [6], ShuffleNet [7] 이 있다. 현재는 DenseNet121, DenseNet169, DenseNet201 까지 학습 및 Evaluation을 완료하였으며, 관련된 결과들에 대해 정리하였다. (본 내용들은 2021년 7월 16일 부터 2021년 7월 24일까지 진행한 내용이다.)

Figure 1 와 같은 순서로 training을 진행하였고, CNN 부분을 ResNet50, DenseNet121, DenseNet169, DenseNet201로 바꾸어가면서 실험을 진행하였다.

A. EXPERIMENTS

Training

Training에 사용된 dataset으로는 MPI-INF-3DHP [8], LSP

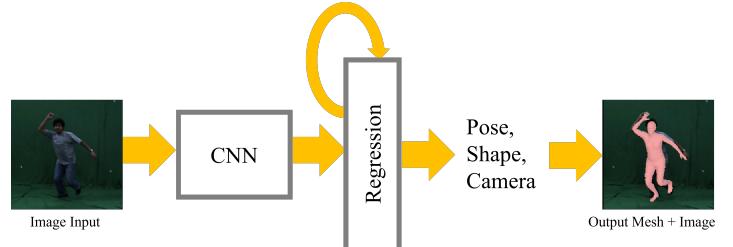


Fig. 1: Overall training step

, LSP-Extended [9], COCO [10], MPII [11]가 있다. 또한 Training은 Geforce RTX 2080 Ti GPU를 사용하였다.

Evaluation

Evaluation에서는 LSP, 3DPW [], MPI-INF-3DHP, H36M [12] dataset을 사용하였다. Metric으로는 가장 많이 쓰이는 mean per joint position error(MPJPE)와 reconstruction error(Rec. Error)를 사용하였다. MPJPE는 모든 joint들의 inference 결과와 ground truth 사이의 거리를 평균하여 얻어지는 값으로 단위가 mm이다.

1) *Experiments-I*: 가장 최근에 나온 좋은 모델들 말고 DenseNet을 먼저 테스트 해 본 이유는 DenseNet의 Architecture가 ResNet에 영감을 받아 만들어졌기도하고, 가장 비슷한 특징이 많다고 생각해서 DenseNet으로 실험을 시작해보았다.

DenseNet121 DenseNet121의 경우 뒷단의 regression step 까지 고려했을 때의 parameter 개수가 3,494,447 개로 매우 작은 편에 속한다. 하지만 이후 결과를 보면 알 수 있지만 학습이 많이 되지 않았던 것을 알 수 있다. 하지만 DenseNet121의 경우 매우 초기에 학습해봤던 것인데, 코드상의 'hassmpl'이라는 값의 issue를 고치기 전인지 후인지 확실하지는 않으나 다른 네트워크보다는 학습이 안 될 것이라 예상했던 것과 같은 결과를 얻었다.

DenseNet169 DenseNet169는 parameter가 총 16,545,797 개 이지만, 이조차도 ResNet50의 공식적인 parameter수 보다도 10,000,000 개 정도 작은 값이다. DenseNet169는 이미 Imagenet에서 ResNet50보다도 좋은 성능을 보였다. 과연, 그 pretrained model을 pose estimation에 그대로 적용했을 때에는 어떤 차이가 있는지 직접 비교해보자하였다. 무조건 model 깊다고 학습이 잘되지는 않듯이 imagenet에서 성능이 좋았다고 해서 3D pose estimation에서도 성능이 좋다는 보장이 없기에 직접 해봐야한다고 생각하였다.

DenseNet201 DenseNet201은 22,410,245개의 parameter를 학습해야하며, 이는 ResNet50보다 조금 작은 값이다. 실제로 Imagenet에서는 DenseNet201이 DenseNet169보다 좋은 성능을 보였기에 ResNet50보다는 parameter수가 작으면서 DenseNet169보다는 Imagenet 성능이 좋은 모델을 테스트 해보자 하였다.

Models	3DPW		MPI-INF-3DHP		H36M-P1		H36M-P2	
	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓
HMR [1]	130	81.3	124.2	-	87.97	58.1	-	56.8
SPIN [5]	96.9	59.2	105.2	-	-	41.1	-	-
ResNet50	129.72	70.66	117.69	78.10	136.08	77.73	148.17	77.57
ResNet50-pruning	130.46	73.31	120.61	81.49	135.22	79.90	147.24	79.67
DenseNet121	273.95	127.44	287.97	142.93	353.41	145.54	282.18	140.70
DenseNet169	121.13	71.37	112.89	76.54	131.12	77.22	137.85	76.17
DenseNet201	133.75	74.77	115.54	76.75	137.76	77.00	152.27	77.47
MobileNetV2	139.00	79.74	129.78	89.40	146.32	86.12	150.74	86.73
MobileNetV3-Large	152.38	84.41	132.53	88.88	163.65	88.16	173.87	88.02
ShuffleNetV2 x1.0	155.16	86.21	152.72	101.93	168.52	96.85	176.94	98.38

TABLE I: Evaluation of models on 3DPW, MPI-INF-3DHP, H36M-P1, H36M-P2. (This table is inspired from VIBE paper [13])

RESULTS and DISCUSSION

TableI에서 알 수 있듯이, DenseNet169보다 DenseNet201이 Imagenet에서의 결과가 좋았지만 오히려 3D human pose estimation에서는 결과가 안 좋은 것을 확인 할 수 있다. 이를 통해 무조건 Imagenet의 결과가 3D human pose estimation에서의 결과로 이어지는 것은 아니라는 것을 알 수 있었다.

3DPW의 경우 SPIN에서만 training 하였고, 나머지 model들의 경우 training 하지 않고 evaluation만 한 결과이다. training을 하지 않은 HMR과 비교해 보았을 때, DenseNet169가 더 성능이 좋았고, 심지어 Discriminator만 제거한 ResNet50과 비교했을 때에도 ResNet50이 성능이 더 좋다는 것을 알 수 있다. Discriminator를 넣은 것보다 안 넣은 것이 결과가 더 좋게 나온 이유가 무엇인지는 더 고민해보고자 한다.

MPI-INF-3DHP에서는 6개의 결과가 다 준수함을 알 수 있다. 여기서도 SPIN 다음으로는 DenseNet169의 결과가 좋은 것을 알 수 있다.

H36M-P1과 H36M-P2 두 데이터 셋의 경우 본 실험에서 training을 하는데 사용하지 않았고, 이로 인해 Error가 크게 나타난 것으로 보인다. 학습이 되지 않았다고 해서 결과가 안 좋은 것은 보안 되어야하는 부분이다. 후에 여러 prior를 사용하였을 때에 개선되는지 잘 살펴봐야하는 부분인 것 같다. 또한 DenseNet121의 경우 아무리 학습이 안되었다고 해도 Error 값이 너무 커서 Evaluation 과정에서 다른 오류가 없었는지 확인해봐야할 것 같다.

Models	Accuracy ↑	F1 ↑	Parts Accuracy ↑	Parts F1 ↑
HMR [1]	91.67	0.87	87.12	0.60
SPIN [5]	91.83	0.87	89.41	0.68
ResNet50	89.95	0.84	87.29	0.62
ResNet50-pruning	89.76	0.84	86.64	0.61
DenseNet121	83.9	0.72	79.92	0.33
DenseNet169	89.9	0.84	87.16	0.61
DenseNet201	90.01	0.84	87.32	0.61
MobileNetV2	87.93	0.80	85.04	0.53
MobileNetV3-Large	88.61	0.82	85.49	0.56
ShuffleNetV2 x1.0	87.10	0.78	84.05	0.50

TABLE II: Evaluation of models on LSP dataset

LSP dataset을 이용한 Evaluation 결과를 TableVI에 정리하였다. 6개의 model 모두 LSP dataset을 이용해서 training 하였기 때문에 성능 비교에 매우 적합하다 생각하였다. LSP에서의 결과는 TableI에서와 다르게 DenseNet201이 DenseNet169보다 성능이 좋았다. 이외의 값들은 SPIN을 제외한 model들의 결과가 매우 비슷했으며, SPIN과도 사실 큰 차이를 보이지는 않았다. 이는 후에 prior를 추가하거나 성능을 더 개선하면 충분히 좋은 결과를 보일 수 있을 것으로 기대한다.

2) *Experiment - II:* 사실 Encoder를 경량화하면서 계속 parameter 수를 살펴보았는데, 실제로는 Encoder 보다 결국 Regression part에서 존재하는 fully connected layer가 많은 parameter 수를 가지고 있었다. 따라서 이부분에서 조금씩 변화를 주어 각 경우들의 성능을 비교해보았다. 실험은 DenseNet169 기반으로 이루어졌으며 뒤에 Regression 부분의 fc2 layer를 빼고 진행한 경우와 fc2 layer 제거와 함께 iteration 수를 줄였을 때에도 어떻게 되는지 비교해보고자 한다.

Results and Discussion

Models	Accuracy ↑	F1 ↑	Parts Accuracy ↑	Parts F1 ↑
HMR [1]	91.67	0.87	87.12	0.60
DenseNet169	89.9	0.84	87.16	0.61
version 1	89.79	0.83	87.08	0.61
version 2	89.58	0.84	86.38	0.61

TABLE III: Evaluation of regression models on LSP dataset

III. CONCLUSION

Encoder 부분만 바꾸어서 실험들을 진행해보고 HMR과 비교해본 결과 엄청나게 큰 차이가 나지 않는 것을 보아, 더 light한 Network도 실험하였을 때 좋은 결과를 기대해 볼 수 있을 것 같다. 이후 남은 기간 동안은 Xception [4], MobileNet [14], ShuffleNet[15] 등 더 다양한 CNN model들을 적용해보고자한다. 또한 현재 ResNet50에 pruning을 적용시켜 training을 진행 중에 있는데 결과를 보고 이후 다른 network에도 pruning을 적용 시켜볼지 결정하게 될 것 같다. 지금까지 실험들은 대부분 training 하는데 20시간에서 24시간 정도가 소요 되었다. 또한 GPU도 대부분의 memory를 사용하였는데, 현재 pruning을 적용한 network는 동일한 batch size임에도 불구하고 사용하고 있는 memory가 훨씬 적은 것을 보아 training이 조금 더 빨라지지 않을까 기대해 본다. 이렇게 다양한 CNN들을 training 하는 동안에는 각각 architecture에 대해 더 자세히 공부하고, 거기서 얻는 아이디어들을 적용 시켜 조금씩 새로운 것들을 해보려한다.

REFERENCES

- [1] A. Kanazawa, M. J. Black, D. W. Jacobs, and J. Malik, “End-to-end recovery of human shape and pose,” in *Computer Vision and Pattern Recognition (CVPR)*, 2018.
- [2] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, “Densely connected convolutional networks,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

Models	3DPW		MPI-INF-3DHP		H36M-P1		H36M-P2	
	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓	MPJPE ↓	Rec. Error ↓
HMR [1]	130	81.3	124.2	-	87.97	58.1	-	56.8
DenseNet169	121.13	71.37	112.89	76.54	131.12	77.22	137.85	76.17
version 1	126.00	71.11	118.03	78.88	130.06	76.65	141.00	76.28
version 2	133.11	72.89	116.53	77.72	134.94	77.70	147.50	76.81

TABLE IV: Evaluation of regression models on 3DPW, MPI-INF-3DHP, H36M-P1, H36M-P2. (This table is inspired from VIBE paper [13])

Models	# Param	3DPW		MPI-INF-3DHP		H36M-P1		H36M-P2	
		MPJPE ↓	Rec Error ↓						
MobileNetV2	5,901,189	139.00	79.74	129.78	89.40	146.32	86.12	150.74	86.73
My MobileNetV2		140.71	78.99	129.99	87.52	151.05	87.03	159.32	87.50
My MobileNetV2 + P1* + FT(20)	3,131,853	144.45	81.23	128.33	85.38	147.65	84.74	160.92	86.43
My MobileNetV2 + P2** + FT(8)		143.39	80.75	124.81	86.16	146.67	87.04	157.68	87.34

Models	Accuracy ↑	F1 ↑	Parts Accuracy ↑	Parts F1 ↑
MobileNetV2	87.93	0.80	85.04	0.53
My MobileNetV2	88.64	0.81	85.82	0.56
My MobileNetV2 + P1* + FT(20)	88.98	0.82	86.14	0.57
My MobileNetV2 + P2** + FT(8)	89.13	0.82	86.29	0.58

TABLE V: Evaluation of regression models on LSP dataset

- [3] G. Huang, Z. Liu, G. Pleiss, L. Van Der Maaten, and K. Weinberger, “Convolutional networks with dense connectivity,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [4] K. S. Lee, “kwotsin/tensorflow-xception: Tensorflow-xception,” May 2017. [Online]. Available: <https://doi.org/10.5281/zenodo.3403277>
- [5] N. Kolotouros, G. Pavlakos, M. J. Black, and K. Daniilidis, “Learning to reconstruct 3d human pose and shape via model-fitting in the loop,” in *ICCV*, 2019.
- [6] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” 2019.
- [7] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “Shufflenet v2: Practical guidelines for efficient cnn architecture design,” 2018.
- [8] D. Mehta, H. Rhodin, D. Casas, P. Fua, O. Sotnychenko, W. Xu, and C. Theobalt, “Monocular 3d human pose estimation in the wild using improved cnn supervision,” in *3D Vision (3DV), 2017 Fifth International Conference on*. IEEE, 2017. [Online]. Available: http://gsv.mpi-inf.mpg.de/3dhp_dataset
- [9] S. Johnson and M. Everingham, “Clustered pose and nonlinear appearance models for human pose estimation,” in *Proceedings of the British Machine Vision Conference*. BMVA Press, 2010, pp. 12.1–12.11, doi:10.5244/C.24.12.
- [10] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, “Microsoft coco: Common objects in context,” in *Computer Vision – ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham: Springer International Publishing, 2014, pp. 740–755.
- [11] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele, “2d human pose estimation: New benchmark and state of the art analysis,” in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3686–3693.
- [12] C. Ionescu, D. Papava, V. Olaru, and C. Sminchisescu, “Human3.6m: Large scale datasets and predictive methods for 3d human sensing in natural environments,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 7, pp. 1325–1339, 2014.
- [13] M. Kocabas, N. Athanasiou, and M. J. Black, “Vibe: Video inference for human body pose and shape estimation,” in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.
- [14] M. Sandler, A. G. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” 2018 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4510–4520, 2018.
- [15] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, “Shufflenet v2: Practical guidelines for efficient cnn architecture design,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.



Fig. 2: The training result of DenseNet169



Fig. 3: The failure case of DenseNet169

Models	Parameters
ResNet50	26,977,501
DenseNet121	2,747,691
DenseNet169	16,545,797
DenseNet201	22,410,245
MobileNetV2	5,901,189
MobileNetV3-Large	7,879,349
ShuffleNetV2 x1.0	4,674,921

TABLE VI: Evaluation of models on LSP dataset