Transfer Learning for Pose Estimation of Illustrated Characters

Shuhong Chen, Matthias Zwicker
{shuhong, zwicker}@cs.umd.edu

UNIVERSITY OF MARYLAND

Winter Conf. on Applications of Computer Vision, WACV 2022

Abstract

Human pose information is critical in many downstream image processing tasks. Likewise, a pose estimator for the illustrated character domain would provide a valuable prior for assistive content creation tasks, such as reference pose retrieval and automatic character animation. We build off prior work by Khungurn et. al., upgrading and expanding their existing ADD illustrated pose dataset, and introducing new datasets for classification and segmentation subtasks. This allows us to perform illustrated pose estimation by efficiently transfer-learning from both domain-specific and task-specific source models. We then apply our resultant state-of-the-art character pose estimator to solve the novel task of pose-guided illustration retrieval. All data and code for our model are available on github.

Challenges from Khungurn et. al.

Limitations of ADD dataset:
- Missing COCO keypoints
- Missing bounding boxes
- Lack of variation

Limitations of transfer methods:
- No transfer from pose task
- Transfer from ImageNet classifier
- Transfer from private 3D data

Our approach

Extend ADD dataset:
- **(A)** Add COCO keypoints
- **(B)** Character segmentation model
- **(C)** +2k hard samples

Effective transfer learning:
- **(D)** MaskRCNN features
- **(E)** Custom Danbooru tagger
- **(F)** All data+code available

Result:

- **(G)** State-of-the-art illustrated pose estimator (+20% PDJ)
- **(H)** Novel application to pose-guided retrieval

References:
Khungurn, P., & Chou, D. (2016, December). Pose estimation of anime/manga characters: a case for synthetic data. MANPU (pp. 1-6).
Lin, T. Y., et. al. (2014, September). Microsoft coco: Common objects in context. ECCV 2014
AniSeg: https://github.com/jerryli27/AniSeg Yet Another Anime Segmenter: https://github.com/zymk9/Yet-Another-Anime-Segmenter
RF5 Danbooru Tagger: https://github.com/RF5/danbooru-pretrained
Anonymous, The Danbooru Community, & Gwern Branwen; "Danbooru2020: A Large-Scale Crowdsourced and Tagged Anime Illustration Dataset", 2020-01-12
Chen, L. et. al. (2017). Rethinking atrous convolution for semantic image segmentation. arXiv preprint arXiv:1706.05587.
Wang, X., Zhang, R., Kong, T., Li, L., & Shen, C. (2020). SOLOv2: Dynamic and fast instance segmentation. arXiv preprint arXiv:2003.10152.
Tian, Z., Shen, C., & Chen, H. (2020). Conditional convolutions for instance segmentation. ECCV 2020
He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. CoRR abs/1512.03385 (2015).
He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2017). Mask r-cnn. ICCV (pp. 2961-2969).

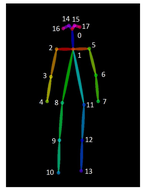
(F) All data + code available at:
`github.com/ShuhongChen/bizarre-pose-estimator`

Artist attributions:
Bounding boxes: background: 日暮竜二 Higurashi Ryuuji (danbooruID 2092753); foreground: ピリジアン (danbooruID 3124941, pixiv user 16083700)
Dataset variation (left-to-right): garnet (danbooruID 1787672, pixiv user 83452); 除け (danbooruID 1852663, pixiv user 101667); 東天紅 (danbooruID 2395348, pixiv user 3757061); このみ (danbooruID 2498174, pixiv user 1933986); UCMM (danbooruID 2635509, pixiv user 89763)
Retrieval row 1 (left-to-right): 露薬 (danbooruID 2467866, pixiv user 263905); Go−1 (danbooruID 3550832, twitter @go_1fk); あれっくす (danbooruID 2686368, twitter @alexmaster55); クラモリ (danbooruID 2690117, pixiv user 1209275); 紅葉(くれは)※お仕事募集中 (danbooruID 2924224, pixiv user 6006540); みの宇 (danbooruID 3159104, pixiv user 1523486)
Retrieval row 2 (left-to-right): 増川 (danbooruID 2407777, twitter @yUiiw); 楠本みや (danbooruID 3410315, pixiv user 6211628); SeNMU (danbooruID 2939081, pixiv user 65308); なもり@ゆるゆり20巻でた (danbooruID 3561420, twitter @_namori_); ratryu (danbooruID 1698324, pixiv user 3892817); ぼや野 (danbooruID 3671428, pixiv user 1263092)
Retrieval row 3 (left-to-right): 九条だんぼ (danbooruID 3314044, twitter @_Dan_ball); 劉祥 (danbooruID 120497, pixiv user 22017); もやL (danbooruID 3669858, twitter @moyashi_mou2); エノキドオ (danbooruID 2803142, pixiv user 4535430); なまもなせ=海通信 (danbooruID 114867, pixiv user 1167548); みぞれまじ (danbooruID 696477, pixiv user 1502612)
Retrieval row 4 (left-to-right): 和果子 (danbooruID 3509383, pixiv user 13748172); アチャコ (danbooruID 1803361, pixiv user 1302618); Seedkeng (danbooruID 2669609, pixiv user 11039166); 雷崎 観雷 (danbooruID 2997827, twitter @yosRRX); テイク (danbooruID 3470661, pixiv user 2096681); いたたたた (danbooruID 2975316, twitter @tatatata6)

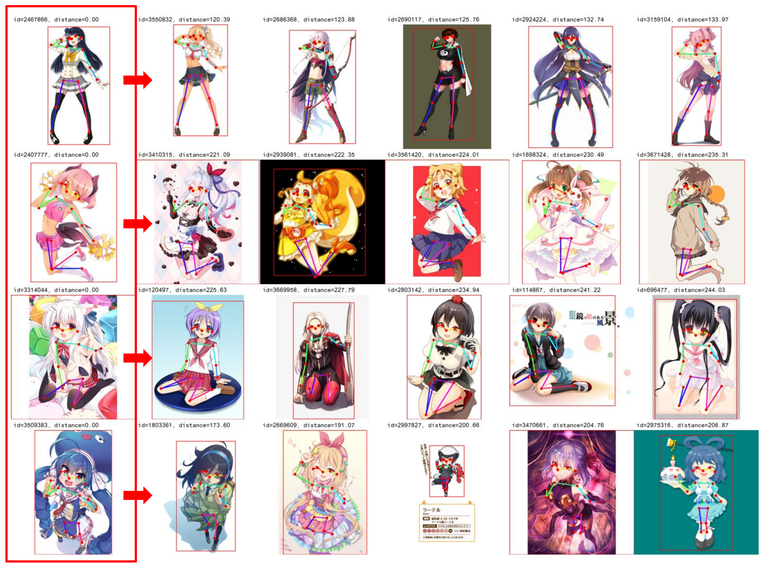
(A) Keypoint standardization

Missing:
- 14: eye_right
- 15: eye_left
- 16: ear_right
- 17: ear_left

Incompatible:
- nose root
- head
- thumbs
- body
- shoe tip



We update ADD (left) to be coco-compliant (right); in particular, we manually label the missing eyes and ears on all 2k samples.



(H) Pose-guided illustration retrieval

Given a query image (left col.), we can extract its pose, and find nearest-neighbor images (rows) based on relative keypoint positions; this may be useful for artists, who often search for references when drawing. Using our model, we can find popular poses such as the wariza (row 3). While our system has no awareness of perspective, it is able to effectively leverage keypoint cues to retrieve similarly foreshortened views in the last row.

(B) Bounding boxes from single-character segmentation

We train a character segmenter, and produce boxes by bounding the predicted segmentations; to train the segmenter, we collected a new dataset of alpha-compositable characters from Danbooru; our segmenter also achieves state-of-the-art single-character segmentation.

Model	F-1	pre.	rec.	IoU
Ours	**0.9472**	**0.9427**	**0.9576**	**0.9326**
YAAS SOLOv2	0.9061	0.9003	0.9379	0.9077
YAAS CondInst	0.8866	0.8824	0.8999	0.9158
AniSeg	0.5857	0.5877	0.5954	0.6651

DeepLabv3 fine-tuned with **20x** larger dataset than AniSeg (filtered from Danbooru)

(C) Increasing dataset variation

We tackle ADD's lack of variation by adding an additional 2k samples (4k total); these were selected from danbooru with difficult back-related tags, such as back, from_behind, looking_back, leaning_back, etc.. Here are some challenging examples added to our dataset.

(D) Efficient transfer learning

We propose a feature concatenation architecture (as well as a more efficient feature matching architecture) that leverages both a task-specific MaskRCNN pose estimator, and an illustration-domain specific classification backbone. Unlike the concatenation model, the matching model does not require the MaskRCNN at inference.

(E) Feature transfer from Danbooru tagger

We use a ResNet50 classification backbone trained on danbooru illustration tags. Instead of using available danbooru taggers like RF5, we use a new manually-filtered tag rulebook, and an inverse square-root reweighing scheme. The reduced class imbalance results in state-of-the-art tagging, and also provides a significant boost to pose estimation.

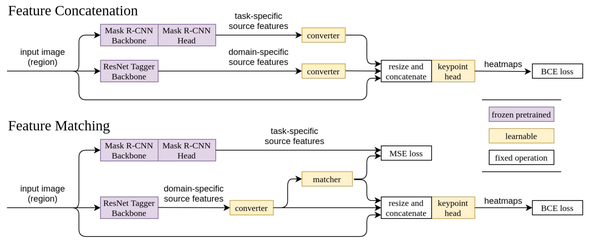
- 1062-class manual tag rulebook
  - 314 body parts
  - 545 clothing types
  - 203 misc.
- 0.38% median class freq. (prev. 0.07%)

$$\mathcal{L}(y,\hat{y}) = \frac{1}{C}\sum_{i=0}^{C-1} w_i(y_i) BCE(y_i,\hat{y}_i)$$

$$w_i(z) = \frac{1}{2}\left(\frac{z}{r_i} + \frac{1-z}{1-r_i}\right)$$

$$r_i = \frac{\sqrt{N_i}}{\sqrt{N_i}+\sqrt{N-N_i}}$$

inverse square-root reweighing scheme



(G) Quantitative results

Due to keypoint incompatibilities, we may only compare PDJ as reported in Khungern et al, and must substitute the closest keypoint for missing datapoints (*). We are able to outperform their model by 10-30% on keypoints aside from “hips”, for which PDJ is a poor metric; please see our paper for more details. We also provide comprehensive ablations of model components in our paper.

tagging performance

Model	Ours	RF5
F-2	0.4744	0.2297
precision	0.3022	0.1238
recall	0.5786	0.3360
accuracy	0.9760	0.9496
F-1	0.4249	0.1910
precision	0.4236	0.1898
recall	0.4458	0.2235
accuracy	0.9851	0.9727

keypoint performance

keypoint	PDJ@20		
nose	0.9897	0.794	(+24.7%)
eyes	0.9928	*0.890	(+11.6%)
ears	0.9795	*0.890	(+10.1%)
shoulders	0.9343	*0.786	(+18.9%)
elbows	0.7916	0.641	(+23.5%)
wrists	0.6961	0.503	(+38.4%)
hips	0.7854	*0.786	(-0.1%)
knees	0.7577	0.610	(+24.2%)
ankles	0.7105	0.596	(+19.2%)
	Ours	Khungurn et. al.	