Journal of Universal Computer Science, vol. 25, no. 6 (2019), 683-700
submitted: 7/1/19, accepted: 20/5/19, appeared: 28/6/19 © *J.UCS*

Synthetic Image Translation for Football Players Pose Estimation

Michał Sypetkowski
(Institute of Computer Science, Warsaw University of Technology, Poland
Sport Algorithmics and Gaming Sp. z o. o., Poland
m.sypetkowski@gmail.com)

Grzegorz Sarwas
(Institute of Control and Industrial Electronics
Warsaw University of Technology, Poland
Sport Algorithmics and Gaming Sp. z o. o., Poland
grzegorz.sarwas@ee.pw.edu.pl)

Tomasz Trzciński
(Institute of Computer Science, Warsaw University of Technology, Poland
t.trzcinski@ii.pw.edu.pl)

Abstract: In this paper, we present an approach for football players pose estimation on very low-resolution images. The camera recording the football match is far away from the pitch in order to register at least half of it. As a result, even using very high resolution cameras, the image area presenting every single player is very small. Additionally, variable weather conditions or shadows and reflections, make this aim very hard. Such images are very hard to annotate by human. In our research we assume lack of manually annotated training data from our target distribution. Instead of manual annotation of large dataset, we create simple python script for rendering synthetic images with perfect annotations. Then we train vanilla CycleGAN (Cycle-consistent Generative Adversarial Networks) for transformation of raw synthetic images into more realistic. We use transformed images to train CPN (Cascaded Pyramid Networks) model. Without bells and whistles, we achieve similar precision on our images as the same CPN model trained with COCO (Common Objects in Context) keypoints dataset.

Key Words: pose estimation, deep convolutional neural network, image translation, synthetic dataset

Category: I.3.3, I.2.10, I.4.8

1 Introduction

Visual analysis of football matches and training sessions is a demanding task, consisting of multiple aspects such as proper video acquisition, tracking in a multi-view system with occlusions, 3D calibration and human behavior analysis. The latter can be split in various conceptual and algorithmic problems, one of each is player's pose estimation. Human pose helps football analysts to validate players' mobility during match and ability to properly perform various game interceptions. In particular analysts check how often player uses non-dominant leg during ball repossession. Accurate pose estimation is also a key step for higher level tasks as analysis of visibility of action for each player, or having an open position while ball pass receiving.

Visual tracking systems installed in football academies uses wide view cameras, spanning on whole pitch or near half. Depending on installation site cameras could be positioned near ground, producing substantial occlusions, or on a high pylons giving non-standard human view from above. Moreover, wide view cameras imposes very low-quality human visuals even for top tier recording hardware. We did not find any literature nor the databases with annotated human pose for the high view and low resolution scenario, what imposed the presented research problem. All successful pose estimation approaches concern high or medium resolution images. The literature presents two generalized approaches in that case. The first one is called bottom-up and the second is top-down. We tested multiple known state-of-the-art algorithms for pose estimation with our custom test images. The images have been acquired form real system with four high-view and high-class wide-view cameras. In next subsections we present analysis of related work in different pose estimation approaches.

1.1 2D multi-person bottom-up approaches

Bottom-up approach predicts all keypoints, which are considered as skeleton model parts in a single scene. Those are further assembled into full skeleton by assigning the parts to appropriate place in the model. In [Cao et al., 2017] the multiple-stage fully convolutional networks for estimating Part Confidence Map (heat map) and PAF (Part Affinity Field) 2D vector field, have been considered. This solution uses multi-stage convolutional network that generates heat map and 2D vector field for each body part (e.g. right elbow, left wrist, neck). The affinity graph is build using 2D vector field part. Based on it, the 2D skeleton with a particular heuristic graph relaxation technique proposed in the article can be constructed. The approach presented in [Lin et al., 2014] achieved the best result in COCO 2016 Keypoint Detection Task, being valid proposition for solving our problem. Along with work of [Simon et al., 2017], this approach has publicly available implementation called OpenPose [Hidalgo et al., 2017]. Highest mAP (mean Average Precision) on MPII multi-person pose dataset [Andriluka et al., 2014] got an approach presented in work of [Newell et al., 2017]. Authors trained a network to simultaneously output detections and group assignments. Output of their neural network consist of detection heatmaps with respective associative embeddings. Grouping body parts is performed by an algorithm based on thresholding the parts embeddings distances. This approach differs from other bottom-up approaches by the lack of separation between detection and grouping. An entire prediction is done at once by a single-stage, generic network based on a stacked hourglass architecture [Newell et al., 2016].

1.2 2D multi-person top-down approaches and single person pose estimation

Top-down approaches localize and crop all persons from an image at first, then solve the single person pose estimation problem (which becomes the main difficulty). Modern single person pose estimation techniques incorporate priors about a structure of human bodies. Best results in COCO 2017 Keypoint Detection Task [Lin et al., 2014] were achieved by CPN (Cascaded Pyramid Network) [Chen et al., 2018]. This algorithm focuses on the "hard" keypoints (i.e. occluded, invisible and with non-trivial background). It is achieved by explicitly selecting the hard keypoints and backpropagating the gradients only from the selected keypoints.

Stacked hourglass [Newell et al., 2016] achieves state-of-the-art result on MPII [Andriluka et al., 2014]. It presents a CNN (Convolutional Neural Networks) architecture for bottom-up and top-down inference with residual blocks. Approach introduced in [Ke et al., 2018] aims to improve stacked hourglass [Newell et al., 2016] achieving the best score on MPII single person pose dataset.

Approach called Mask R-CNN (Mask Regions with CNN features) [He et al., 2017], extends Faster R-CNN [Ren et al., 2017] by adding a branch for predicting an object mask in parallel with bounding box recognition. Using this simple modification, Mask R-CNN can be applied to keypoints detection. This approach achieves high AP in all COCO 2017 challenges (i.e. object detection, object segmentation, keypoint detection).

[Simon et al., 2017] presents precise hand 2D keypoint detector. It introduces a semi-supervised training algorithm called Multiview Bootstrapping. Initially, the algorithm needs a set of annotated examples. The model is trained using only these examples at the beginning. Then, the model detects keypoints on unannotated examples with multiple camera views. Each multi-view example is then robustly 3D triangulated, and reprojected creating additional training set.

1.3 Other approaches

Modern pose estimation approaches are already robust to blurring and low-resolution in general. Significantly improving their performance with simple methods, like heuristic data augmentation or upscaling the images with generic upscaling algorithms may be extremely hard with limited training data.

A straightforward solution for improving the results on images from a specific distribution may be manual annotation of some examples (e.g. a few thousands) for training or fine-tuning existing state-of-the-art models. Manual annotations on low resolution images not only require immense amount of work, but also may be hard to be done precisely in our case.

In our approach we consider single person pose estimation on large dataset of blobs detected with external tracking system. We create synthetic dataset and improve it using modern achievements of GANs (Generative Adversarial Networks). In the following sections, we discuss selected existing approaches that use synthetic dataset, and use of GANs for pose estimation.

2 Synthetic datasets

In this section we present successful approaches that focus on generating large (practically infinite, but every distribution have it's effective variety limit) annotated synthetic or partially synthetic (e.g. [Dwibedi et al., 2017]) datasets with minimal effort. They show that limited realism may provide enough training signal for current state-of-the-art object/keypoints detector models.

In [Dwibedi et al., 2017] to create the dataset authors propose simply 'cut' real object instances and 'paste' them on random backgrounds (without any perspective or lighting adjustment). This process implemented in naive way would give the trained model possibility of exploiting subpixel discrepancies at the boundaries. To address this problem, the approach blends 'cut' objects into the background with heuristic methods. Additionally, it blends in the same way the distractor objects along with the correct ones. Synthetic data is then feed into the model along with the real data. In the end, such training set gives significantly higher performance, than the non-augmented dataset. (e.g. 51 AP instead of 42 AP on GMU Kitchen Scenes [Georgakis et al., 2016] dataset).

In [Gupta et al., 2016] Gupta et al. introduce a method of 'pasting' synthetically rendered text into the real images with respect to the local region cues, i.e. surface geometry predicted with other models and local colors. Models trained with such dataset achieve high accuracy in the task of text detection in the wild.

In [McCormac et al., 2016] authors create fully synthetic dataset using ray-trace rendered scenes — interiors of buildings. It shows that large-scale high-quality synthetic RGB datasets with task-specific labels can be more effective for pre-training than the large-scale real-world images dataset like ImageNet [Deng et al., 2009].

Many modern successful approaches concerning synthetic dataset use in some way real images to create it:

- [Dwibedi et al., 2017] uses both real examples and synthetic. Synthetic images are made by simple editions of the real images, therefore they are not dependent on graphics renderings.
- [Mueller et al., 2018] uses real examples (unpaired with synthetic) for improving the distribution of 3D rendered synthetic images.

– [Gupta et al., 2016] 'pastes' rendered text into the real images (background).

– [McCormac et al., 2016] does not use real images explicitly, but the 3D models of furniture may be using textures that are created at some degree using real photos.

In paper [Dwibedi et al., 2017] authors suggest that state-of-the art detection methods like Faster R-CNN [Ren et al., 2017] care more about local region-based features for detection than the global scene layout. This fact somehow justifies their result.

In our previous article [Sypetkowski et al., 2019] we have shown robustness to low resolution and small distortions, of CPN [Chen et al., 2018] trained on large datasets. One may suspect, that in our case artifacts of rendered synthetic dataset will not cause the optimizer to find significant exploits, or stuck in a bad local minimum.

3 Generative Adversarial Networks for image generation and pose estimation

In recent years, we observe a rapid progress in results achieved by Generative Adversarial Networks for image generation. In this section, we review selected approaches and discuss application of GANs in pose estimation task.

Initial research concerning image generation using GANs was done by [Goodfellow et al., 2014]. In recent years, there was many improvements for loss functions, model architecture and overall training process (DCGAN [Radford et al., 2015], LSGAN [Mao et al., 2017], SRGAN [Ledig et al., 2017], StackGAN [Zhang et al., 2017], Wasserstein GAN [Arjovsky et al., 2017], Improved Wasserstein GAN [Gulrajani et al., 2017]). Modern state-of-the art approaches (StackGANv2 [Zhang et al., 2018], Progressive growing of GANs [Karras et al., 2017]) can infer photo-realistic high-resolution images using multi-stage generator architecture. Recent paper [Karras et al., 2018] introduced an architecture that allows unsupervised separation and control of high, mid, and low-level attributes of high-resolution, photo-realistic generated images.

Adversarial PoseNet [Chen et al., 2017] presents an interesting approach that trains a GAN, with multi-task pose generator and two discriminator networks. It achieves state-of-the-art results on MPII [Andriluka et al., 2014] single person pose estimation dataset. The model consists of the generator network, the pose discriminator network and the confidence discriminator. Half of generated heatmaps represent keypoint locations and the other half occlusion predictions. The generator architecture is based on stacked hourglass architecture [Newell et al., 2016].

In pose estimation from low-resolution images, an idea worth consideration is generative upscaling. Modern generic upscaling deep learning methods are focused on minimizing MSE (Mean Squared Reconstruction Error) [Dong et al., 2016]. SRGAN (Super Resolution GAN) [Ledig et al., 2017] is capable of inferring photo-realistic natural images for 4x upscaling factors. The approach uses GAN, trained using a perceptual loss function consisting both of an adversarial loss and a content loss. Such generic upscaling algorithms like these will not improve results on our dataset as we have shown in our preliminary article version [Sypetkowski et al., 2019], because of too low resolution and characteristic distortions caused during the scene recording (e.g. compression). Super-FAN [Bulat and Tzimiropoulos, 2018] addresses the problem of generative upscaling of very low resolution images. It focuses on improving the quality of low resolution facial images and locating the facial landmarks on such images. The idea is to connect third network (Face Alignment Network) to GAN. This third network detects facial landmarks on the upscaled image. Generator loss includes additional component – landmark detection loss. Therefore it learns to generate face that fits geometrically.

3D Hand pose estimation approach [Mueller et al., 2018] focuses on enhancing a synthetic dataset to make their distribution more like the distribution of the real images. It uses CycleGAN with an additional geometric consistency loss. The paper shows, that training with generated images significantly outperform standard augmentation techniques. Similar approach may be applied to pose estimation.

In our approach we propose to use CycleGAN for enhancing synthetic dataset for pose estimation. Our images are very low-resolution and the human body details are not visible on our images, therefore GANs are easier to learn their distribution.

4 Proposed approach

For our tests we gathered data using 4 cameras placed at the field corners. Because of resolution limits, in practice we can assume that for a given player only 2 cameras are close enough to produce usable visuals. The cameras are production class CCTV devices with 4K resolution and high compression bandwidth. Even though the crop factor around single player magnifies compression and optics artifacts, which renders high frequency data unusable. Low quality and viewing angle creates uncommon characteristics of the images. Comparing this scenario with the standard pose estimation datasets like COCO [Lin et al., 2014] and MPII [Andriluka et al., 2014] we can list main problems:

- Human based annotations are much more difficult and time consuming for our images. Some images have practically indistinguishable joint locations, even with much human time and effort spent

- Border areas of the pitch generates almost top down views, where the human parts are mostly occluded by upper body
- Images are blurred with non-deterministic distribution, which makes generic upscaling algorithms useless
- All players wear single-color clothes, which makes it harder to distinguish limbs (especially hands) from the body

In our preliminary article [Sypetkowski et al., 2019] we selected most efficient network architecture trained with external data (see Table 1). In all experiments, we used CPN model [Chen et al., 2018] (smaller version – with input resolution of 256x192 and based on ResNet50 [He et al., 2016]).

Our new approach consists of 4 steps:

1. rendering synthetic dataset (see Section 4.1),
2. training CycleGAN [Zhu et al., 2017], using generated synthetic dataset as first distribution examples, and real players blobs for the second (see Section 4.2),
3. training CPN [Chen et al., 2018], with synthetic dataset, cycled-synthetic dataset, and mixed with COCO [Lin et al., 2014] dataset,
4. measuring pose estimation accuracy on our benchmark (see Section 4.3).

4.1 Rendering synthetic dataset

We use Blender[1] for scene modelling and rendering, and ManuelbastioniLAB[2] for creating human 3D models. We use blender ray-trace rendering engine – cycles. We design armature pose distribution empirically – by randomizing bones IK (Inverse Kinematics) targets transform (with respect to the rest pose - A-pose) with normal distributions. One character armature has 8 IK targets in total: 2 for hands, legs, elbows, 1 for body center and a head look-at position. Each IK target, has hard-coded means and standard deviations for each axis, e.g.:

- hands IK targets have higher standard deviation on backward-forward axis than on left-right axis, because hands are moving usually switching between front and back position during running
- similarly feet IK targets have higher standard deviation on backward-forward axis than on left-right axis, because running is usually for forward movement

[1] `https://www.blender.org/`
[2] `http://www.manuelbastioni.com/`

– mean of body center IK is lower than in the rest pose, because it is usually lower during dynamic actions like running or kicking

We randomize each IK target independently. Additionally, we constraint ran-

Approach	**Implementation / experiment**	**Training set**	**Language Library**	**corr. pose**	**corr. legs**	**N /A**
PAF	OpenPose[1]	COCO	C++, Caffe	58	106	29
Stacked hourglass	original implementation[2], 8-stack model	MPII	Lua, Torch	142	203	-
	alternative implementation[3], hg_refined_200, 4-stack model	MPII	Python, Tensorflow	29	90	
	alternative implementation - not official[4], 8-stack model	MPII	Python, Pytorch	135	186	
CPN	original implementation[5], COCO.res50.256x192, snapshot_350.ckpt	COCO	Python, Tensorflow	171	**224**	-
	SRGAN for upscaling			90	167	
	blurred images, 50 more epochs, lr 1.6e-5 (from COCO.res50.256x192)			155	206	
	COCO.res101.384x288, snapshot_350.ckpt			158	223	

Table 1: Selected human pose estimation implementation results (original and our experiments). The table contains results of experiments from our previous paper. Measured implementations vary in skeleton structure used as a reference, therefore the measurements are done without annotated testset. We've taken into account few the easiest football aspects for automation. We measured precision on 300 test images with human based decision, whether the answer is one of 4 classes: correct, only correct legs pose estimation, wrong pose, N/A. The human-based bias has been lowered by cross-checkup with industry football analyst but still may produce significant variance, opposed to keypoint-based difference metrics.

[1] `https://github.com/CMU-Perceptual-Computing-Lab/openpose`
[2] `https://github.com/umich-vl/pose-hg-demo`
[3] `https://github.com/wbenbihi/hourglasstensorlfow`
[4] `https://github.com/bearpaw/pytorch-pose`
[5] `https://github.com/chenyilun95/tf-cpn`

domized pose with heuristics:

- in general, feet are standing on the ground (jumping positions are rare)
- if left foot is moved forward and standing on the ground, then probably right knee is bent backwards (as it is during the run)
- foot in the air (not standing on the ground) is usually rotated similarly to the corresponding calf
- heel may be lifted when leg is standing

We rendered 1000 football fields with constant camera position, various lighting angle, and randomly (with uniform distribution) placed and rotated 100 players. Each image has resolution 4000x3000 (like the original cameras). Then, we cut 100k training blobs from the large images. Example synthetic blobs are shown in Fig. 1.

4.2 CycleGAN-ing synthetic dataset

In experiments we train vanilla CycleGAN [Zhu et al., 2017] architecture with 256x256 input / output. We use 100k synthetic blobs for first distribution, and 186K real blobs from 2.1k captured sequences for the second distribution. Example CycleGAN-ed training images are shown in Fig. 2. The model is trained with batch size of 1.

4.3 Benchmark

We annotated 400 real blobs with full skeletons. First, all testing blobs are fit into 256x192 rectangles. For each blob we measure OKS (Object Keypoint Similarity) given by:

$$OKS = \frac{\sum_i^n exp(-d_i^2)}{n}, \tag{1}$$

where

$$d_i = \sqrt{\left(\frac{x_i - \overline{x_i}}{32}\right)^2 + \left(\frac{y_i - \overline{y_i}}{32}\right)^2},$$

$x_i,\ y_i$ are ground truth keypoint coordinates in pixel space,

$\overline{x_i},\ \overline{y_i}$ predicted coordinates in pixel space,

n is number of keypoints (in our case it is 13).

In COCO human keypoint annotations, head has 5 defined keypoints (eyes, ears and nose). For our benchmark we merge it into one keypoint (averaging

Figure 1: Example synthetic blobs with drawn ground truth skeletons.

coordinates of these 5 keypoints, both in prediction and ground truth), because such details are not visible at all on our images. In the end, we consider 13 keypoints (the network predicts 17). We measure mean OKS over all test blobs. We consider our testset sufficiently large for measurement (see Fig. 3).

Additionally we create raw synthetic benchmark on raw synthetic images to better illustrate capabilities of our trained models. This testset consists of 10k blobs (it is not included CycleGAN training set).

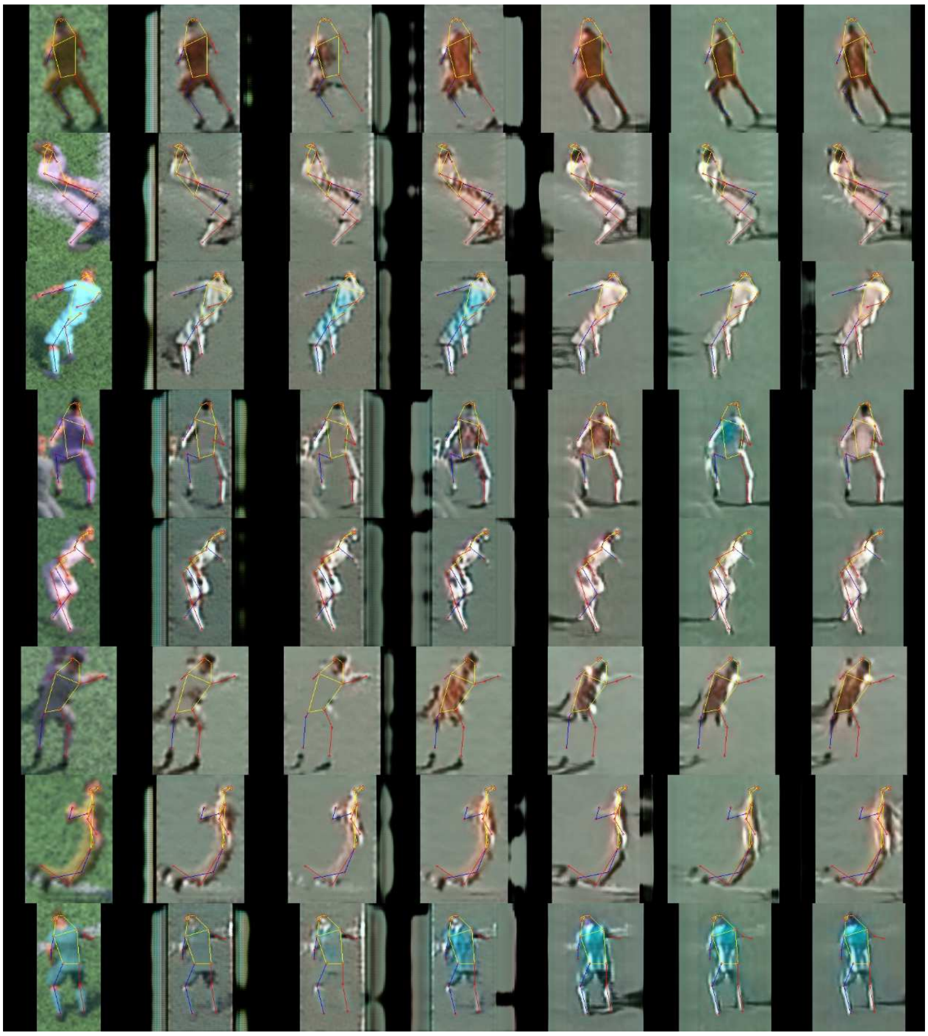

Figure 2: Example images from CycleGAN with drawn ground truth skeletons. The skeletons are drawn with thin lines, so that visual artifacts are visible. First column shows original synthetic images, others correspond to training iterations – from left: 5k, 10k, 15k, 50k, 100k, 180k. Transformed images are not perfectly consistent geometrically, but characteristic distortions occurring in the real conditions are performed in appropriate parts of the image – it enables the network comprehensible inference on the real images.

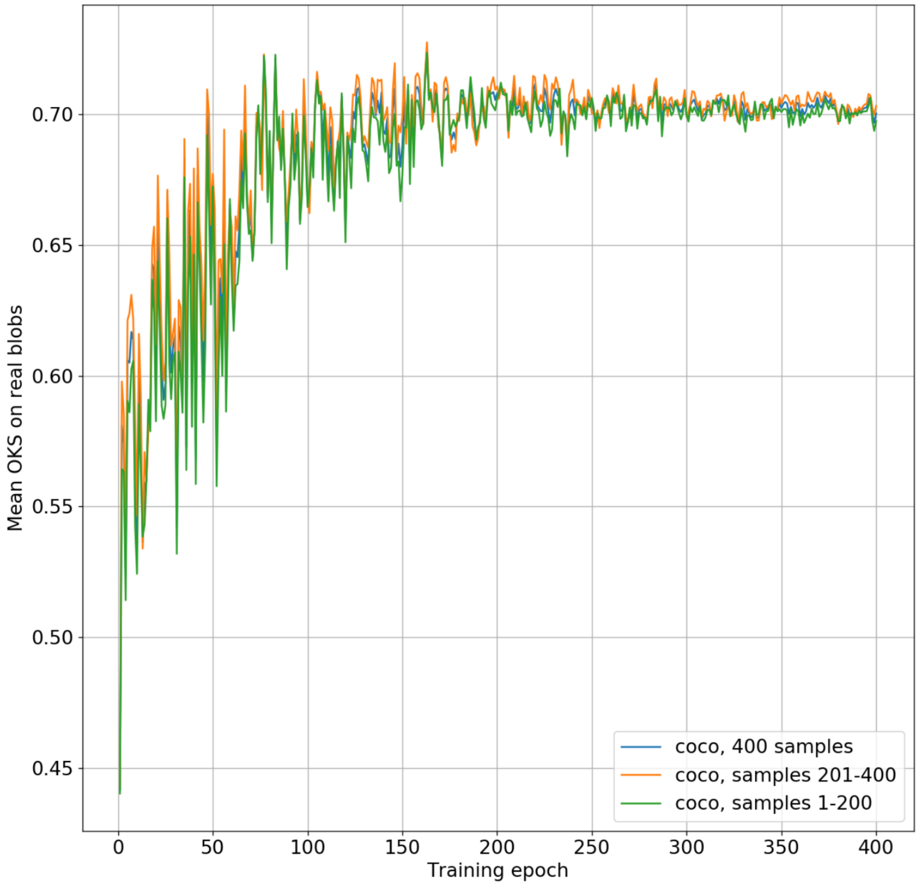

Figure 3: The same model trained on COCO gives similar plot shape when evaluated on 2 separated parts of our small testset. Basing on this observation, we assume that our benchmark allows meaningful comparison between different models.

5 Experiments results

In experiments, we use CycleGAN-ed images made using checkpoint after 5k, and 18k iterations of training CycleGAN. As source raw synthetic images for transformation we use the same 100k samples that were used during the training. Despite of our efforts to annotate our testset precisely, it still contains many incorrect annotations – captured images are of very low quality. Therefore, our benchmark shows only the major differences between tested approaches. Example detections for selected experiments are shown in Fig. 5. Mean OKS value of these experiments over training epoch is shown in Fig. 4. Because of different train and evaluation data distributions, the plots are noisy. Still, as we discussed testset sufficiency in Section 4.3, the plots (including the noise) are meaningful. COCO Minival and raw synthetic benchmark scores for selected checkpoints are shown in Table 2.

Training set	Training epoch	Mean OKS (our benchmark)	Mean OKS (our synthetic benchmark)	COCO (Minival) AP @0.5:0.95
COCO	163 (best)	**0.725**	0.824	0.691
COCO	400	0.700	0.836	**0.700**
COCO+CycleGANed	46 (best)	**0.725**	0.926	0.595
CycleGANed	23 (best)	0.691	0.923	0.009
Raw synthetic	5 (best)	0.572	0.966	0.006
Raw synthetic	100	0.303	**0.977**	0.004

Table 2: Summary of selected checkpoints scores. Epoch marked with "(best)", is the one after which the model achieves best score (among the other checkpoints from this experiment) on our real images benchmark.

Usually, best score is achieved in early epochs of training. It is possible, because our training and evaluation images are from different distributions.

Clearly, long training with only raw synthetic data causes the model to exploit synthetic artifacts and assumptions based on imperfect artificial heuristic pose distribution. In this case the model learns artificial distribution very easily – achieves almost perfect results on this distribution after only 5 epochs (see Table 2).

Training with CycleGAN-ed data achieves high mean OKS (close to training on COCO) in early training epochs, therefore our augmentation method of raw synthetic dataset makes its distribution more similar to the real images distribution. Training with CycleGAN-ed images from early checkpoint (5k iterations) shows somewhat averaged results between training with raw synthetic and later checkpoint (18k iterations) CycleGAN-ed dataset. In general, models trained with artificially created (for our domain) data doesn't work at all on COCO benchmark – their score is close to 0. Moreover, mixed dataset training decreases the score.

Despite our experiments are not exhaustive (e.g. in this paper we try only one

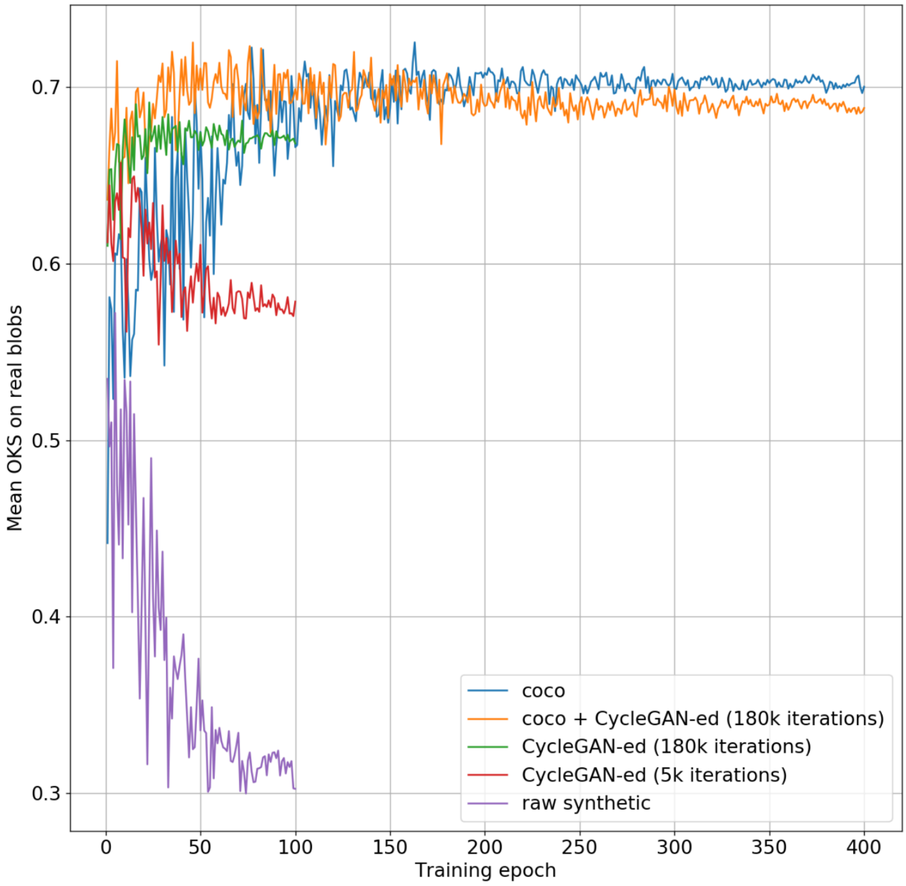

Figure 4: Mean OKS over training epoch. Initial learning rate is 0.0005, and it is decreased by 50% after each 60 epochs in experiments where COCO dataset is used, and after 15 epochs in the other experiments. We use batch size of 32. Each epoch is 60k randomly sampled images without returning (with original CPN data augmentation).

option in terms of selecting various model parameters), mixed dataset training achieves high scoring checkpoints on our benchmark faster than training on COCO only. We suspect that detailed experiments on various stages of our experiments may achieve even higher mean OKS without much effort. Research in this direction would first require creating a better benchmark – larger, and with more precise annotations.

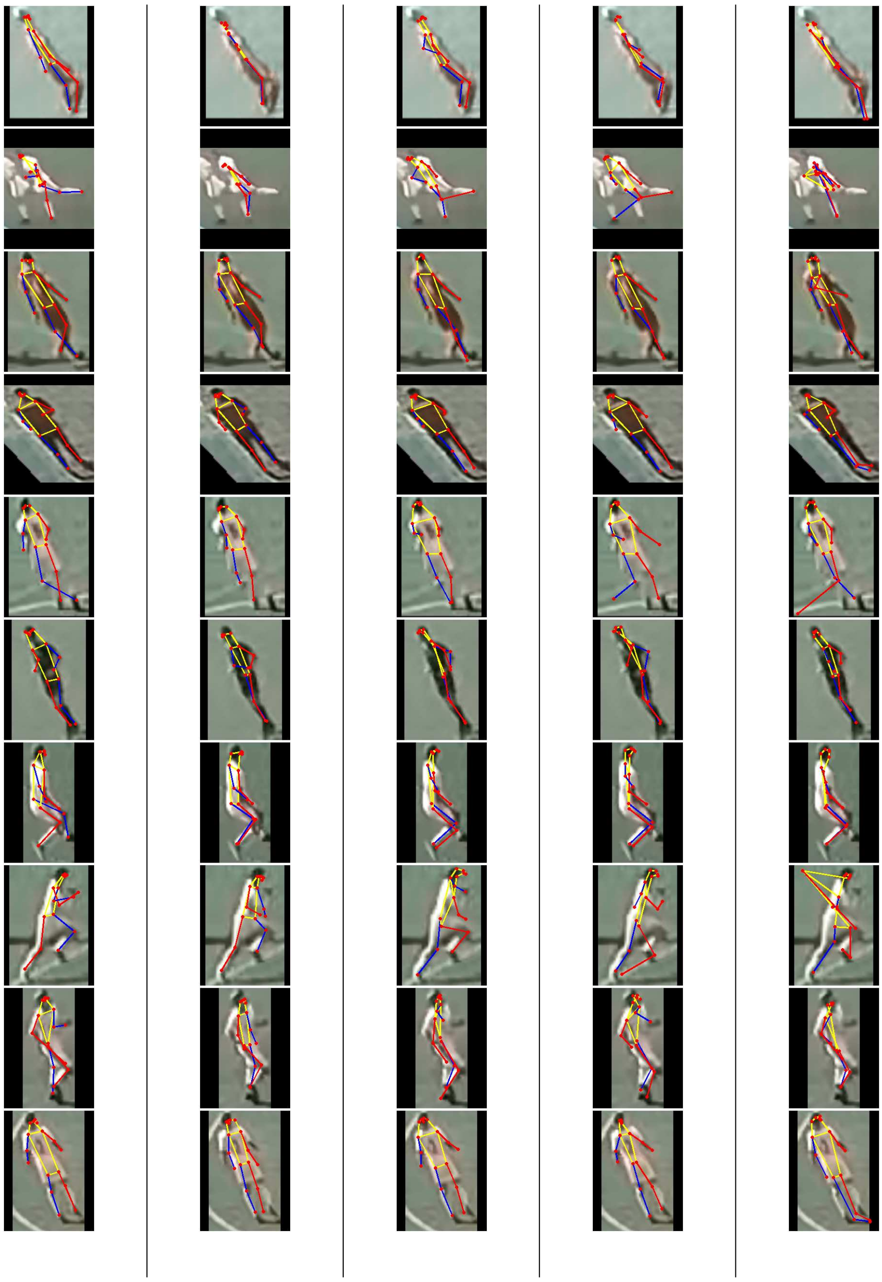

Figure 5: Example pose estimations for best checkpoints. Columns from left: ground truth, coco, coco + CycleGAN-ed, only CycleGan-ed, raw synthetic.

6 Conclusions

In this paper, we focus on football players pose estimation on very low-resolution images, received from the actual High Quality CCTV system located on lighting spots in the corners of the football pitch. To omit the need for the manual annotation of many thousands of training examples we create simple python script for rendering synthetic images. In order to give more realism to our raw synthetic images we used vanilla CycleGAN. Conducted experiments proved, that training neural networks for pose estimation without manually annotated data can (in some cases) achieve as good results as training with large, manually annotated, generic datasets (like COCO keypoints). With more exhaustive experiments, it may be possible to achieve even better results by changing synthetic dataset generation method, various hyperparameters, and architectures of both image translation and pose estimation models.

Precise annotation of a large training set requires many hours of human labor, while script for rendering synthetic dataset for a specific task using heuristics, can be created by one person and much faster.

Acknowledgements

This work was co-financed by the European Union within the European Regional Development Fund grant no. POIR.01.02.00-00-0153/17-00.

References

[Andriluka et al., 2014] Andriluka, M., Pishchulin, L., Gehler, P., and Schiele, B. (2014). 2D human pose estimation: New benchmark and state of the art analysis. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 3686–3693.

[Arjovsky et al., 2017] Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223.

[Bulat and Tzimiropoulos, 2018] Bulat, A. and Tzimiropoulos, G. (2018). Super-fan: Integrated facial landmark localization and super-resolution of real-world low resolution faces in arbitrary poses with gans. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pages 109–117.

[Cao et al., 2017] Cao, Z., Simon, T., Wei, S. E., and Sheikh, Y. (2017). Realtime multi-person 2D pose estimation using part affinity fields. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 1302–1310.

[Chen et al., 2017] Chen, Y., Shen, C., Wei, X. S., Liu, L., and Yang, J. (2017). Adversarial posenet: A structure-aware convolutional network for human pose estimation. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 1221–1230.

[Chen et al., 2018] Chen, Y., Wang, Z., Peng, Y., Zhang, Z., Yu, G., and Sun, J. (2018). Cascaded pyramid network for multi-person pose estimation. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pages 7103–7112.

[Deng et al., 2009] Deng, J., Dong, W., Socher, R., Li, L., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pages 248–255.

[Dong et al., 2016] Dong, C., Loy, C. C., He, K., and Tang, X. (2016). Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307.

[Dwibedi et al., 2017] Dwibedi, D., Misra, I., and Hebert, M. (2017). Cut, paste and learn: Surprisingly easy synthesis for instance detection. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 1310–1319.

[Georgakis et al., 2016] Georgakis, G., Reza, M. A., Mousavian, A., Le, P., and Košecká, J. (2016). Multiview rgb-d dataset for object instance detection. In *Proc. Fourth Int. Conf. 3D Vision (3DV)*, pages 426–434.

[Goodfellow et al., 2014] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

[Gulrajani et al., 2017] Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., and Courville, A. C. (2017). Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777.

[Gupta et al., 2016] Gupta, A., Vedaldi, A., and Zisserman, A. (2016). Synthetic data for text localisation in natural images. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 2315–2324.

[He et al., 2017] He, K., Gkioxari, G., Dollár, P., and Girshick, R. (2017). Mask r-CNN. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 2980–2988.

[He et al., 2016] He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

[Hidalgo et al., 2017] Hidalgo, G., Cao, Z., Simon, T., Wei, S.-E., Joo, H., and Sheikh, Y. (2017). Openpose. https://github.com/CMU-Perceptual-Computing-Lab/openpose.

[Karras et al., 2017] Karras, T., Aila, T., Laine, S., and Lehtinen, J. (2017). Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*.

[Karras et al., 2018] Karras, T., Laine, S., and Aila, T. (2018). A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*.

[Ke et al., 2018] Ke, L., Qi, H., Chang, M., and Lyu, S. (2018). Multi-scale supervised network for human pose estimation. In *Proc. 25th IEEE Int. Conf. Image Processing (ICIP)*, pages 564–568.

[Ledig et al., 2017] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., and Shi, W. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 105–114.

[Lin et al., 2014] Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

[Mao et al., 2017] Mao, X., Li, Q., Xie, H., Lau, R. Y. K., Wang, Z., and Smolley, S. P. (2017). Least squares generative adversarial networks. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 2813–2821.

[McCormac et al., 2016] McCormac, J., Handa, A., Leutenegger, S., and Davison, A. J. (2016). Scenenet rgb-d: 5m photorealistic images of synthetic indoor trajectories with ground truth. *arXiv preprint arXiv:1612.05079*.

[Mueller et al., 2018] Mueller, F., Bernard, F., Sotnychenko, O., Mehta, D., Sridhar, S., Casas, D., and Theobalt, C. (2018). Ganerated hands for real-time 3D hand tracking from monocular rgb. In *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*, pages 49–59.

[Newell et al., 2017] Newell, A., Huang, Z., and Deng, J. (2017). Associative embedding: End-to-end learning for joint detection and grouping. In *Advances in Neural Information Processing Systems*, pages 2277–2287.

[Newell et al., 2016] Newell, A., Yang, K., and Deng, J. (2016). Stacked hourglass networks for human pose estimation. *Computer Vision – ECCV 2016*.

[Radford et al., 2015] Radford, A., Metz, L., and Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*.

[Ren et al., 2017] Ren, S., He, K., Girshick, R., and Sun, J. (2017). Faster r-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137–1149.

[Simon et al., 2017] Simon, T., Joo, H., Matthews, I., and Sheikh, Y. (2017). Hand keypoint detection in single images using multiview bootstrapping. In *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR)*, pages 4645–4653.

[Sypetkowski et al., 2019] Sypetkowski, M., Kurzejamski, G., and Sarwas, G. (2019). Football players pose estimation. In Choraś, M. and Choraś, R. S., editors, *Image Processing and Communications Challenges 10*, pages 63–70, Cham. Springer International Publishing.

[Zhang et al., 2017] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. (2017). Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *Proc. IEEE Int. Conf. Computer Vision (ICCV)*, pages 5908–5916.

[Zhang et al., 2018] Zhang, H., Xu, T., Li, H., Zhang, S., Wang, X., Huang, X., and Metaxas, D. N. (2018). Stackgan++: Realistic image synthesis with stacked generative adversarial networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, page 1.

[Zhu et al., 2017] Zhu, J.-Y., Park, T., Isola, P., and Efros, A. A. (2017). Unpaired image-to-image translation using cycle-consistent adversarial networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2223–2232.