SkateMAE: Synthetic Data for Skateboard Pose Estimation

Yau-Meng Wong
Hong Kong University of Science and Technology
ywongar
ywongar@connect.ust.hk

Abstract

This work presents a simple approach for a novel task - skateboard pose estimation. The application of AI to sports is a growing field, for example Google's TacticAI helps Liverpool FC plan corner kick formations. Skateboarding is also growing, appearing in the Olympics for the first time in 2021. Object pose estimation is an obvious task to apply to skateboarding, as the rotation of the board is how tricks are generally defined. Due to the absence of large-scale annotated datasets for skateboard poses, I propose a synthetic data generation pipeline. A 3D skateboard model is rendered with randomized camera poses, and the rendered images are composited with real images to create realistic training data. A Masked Autoencoder (MAE) model is trained on the synthetic data to predict camera distance, elevation, and azimuth from single images. While the model performs well on the synthetic test set, its performance on real data is hindered by the domain gap between synthetic and real images. To address this challenge, data augmentation techniques are employed. Promising quantitative results are observed on the synthetic test set, and slight qualitative improvement on the real data, but there is still some way to go. Future directions include incorporating temporal information, generating synthetic video sequences, and exploring alternative models. Overall, this work demonstrates the potential of synthetic data and domain adaptation techniques for skateboard rotation estimation.

1. Introduction

Skateboarding has emerged as a popular sport in the last few decades, attracting a significant following and even making its debut in the Olympics in 2021. With the increasing application of artificial intelligence (AI) to various sports domains [10][6], leveraging AI techniques for skateboarding-related tasks emerges as a potential niche. I chose skateboard pose estimation, accurately determining the rotational orientation of the skateboard, because this information is crucial for classifying and understanding skateboarding tricks. Skateboard tricks on flat ground are generally defined by the rotation of board and the rotation of the skateboarder. In this paper I focus on the rotation of the board only, as existing human pose estimation frameworks are already capable of determining the skateboarder's pose. These frameworks could be used in parallel with this work for a trick classification model in the future.

Achieving accurate skateboard pose estimation poses several challenges. One major obstacle is the lack of large-scale annotated datasets specifically tailored for skateboard poses. Traditional object pose estimation methods heavily rely on ground truth data for training, which is difficult to obtain for skateboarding due to the absence of comprehensive datasets. Consequently, there is a need to explore alternative approaches to generate training data for skateboard pose estimation.

In this work, I propose a novel approach that utilizes synthetic data generation to address the data scarcity issue. The synthetic data generation pipeline involves rendering a 3D skateboard model from randomized camera angles. These rendered images are then composited with real-world images, creating a hybrid dataset that captures the realism and diversity of skateboard poses. By leveraging this synthetic dataset, a model can accurately predict the camera distance, elevation, and azimuth from single images, thereby enabling robust skateboard rotation estimation.

The core of our approach lies in the utilization of a Masked Autoencoder (MAE) model [7], which is pretrained on both real and synthetic data. The MAE model is designed to reconstruct images while simultaneously extracting meaningful latent representations. From this latent representation a classification head can predict the camera parameters associated with the skateboard pose. However, a significant challenge arises when attempting to deploy the trained model on real-world data. The domain gap between synthetic and real images leads to low generalization capabilities.

To bridge this domain gap and enhance the model's performance on real data, we employ data augmentation techniques. Specifically, I overlay masks from real images,

along with random color jitter and image translations along the horizontal and vertical axes. These techniques aim to reduce the semantic differences between the synthetic and real images, thereby improving the model's ability to generalize to real-world scenarios.

2. Previous Work

There are only a few previous instances of AI applied to skateboarding. They have generally focused on end-to-end trick classification, with monocular video [3][5] or sensor data [1]. All of the proposed models are trained on a set list of trick classes, which far from encompass the theoretically infinite sum total of possible tricks. To teach one of these models a new trick would require collecting additional labelled real world data and retraining the model. This is one of the main reasons I focused on board rotation estimation. A model that can accurately predict board rotation (along with human pose for the skater) would be able to generalize to any flat ground trick, even if it wasn't present in the training data. Although this vision of generalizable trick classification isn't realized in this paper, it's on the horizon.

3. Data Collection

3.1. Real World Data

I initially began collecting real world data from Youtube. I decided to constrain the scope of the data to single, short (less than 5 second) clips of flat ground tricks. All the real world data was downloaded from the channel "The Berrics", specifically their "Battle at the Berrics" series. I chose these videos as they are quite structured, so clips can be scraped efficiently, and limited to flat ground tricks only. These real world clips were labelled with the type of trick being done, though this data wasn't used in this work.

After compiling 60 video clips, I used Meta's DETR model with a Resnet-50 backbone pretrained on COCO [2][8] to generate bounding boxes for the skateboard in each frame (thankfully COCO includes "skateboard" as an object class). I used these bounding boxes to crop and resize each frame to 128x128 images of the skateboard (see Figures 3 and 4).

3.2. Synthetic Data

To generate synthetic data for rotation estimation I used Pytorch3D [9] to render a 3D skateboard model, and saved the randomized camera distance, elevation, and azimuth data to use as ground truth data for training. The initial render contains the skateboard on a white background, which doesn't quite match what the real world data looks like, so I went about reducing this domain gap. The first thing I did was overlay the rendered skateboard onto a background image cropped randomly from the real video frames. This occurred during training, during which background images were chosen at random for each epoch, so it effectively acted as a data augmentation.

Figure 1. Synthetic image example. Skateboard is rendered on top of a background image cropped from real data, and the "leg mask" created with MaskFormer [4] is overlaid on top. In both pretraining and training the "leg mask" and backgrounds are randomized, along with additional more classical augmentations.

This was a good start, but there was one glaring difference between the real and synthetic images: the real images contained the skateboarder. This is an important fact for pose estimation, as the skateboarder's feet often occlude the skateboard, making the task more complicated for the model to learn. To solve this I used Meta's MaskFormer [4] model pretrained on COCO for semantic segmentation to create cutouts of the skater's legs and overlaid them on top of the synthetic images. In my ablation study this change significantly improved model performance on the synthetic test set.

Along with these augmentations, I used random translation and color jitter to reduce overfitting on the synthetic data. The random translation was possible because I've simplified the task of pose estimation to just rotation, not translation, so it didn't obstruct any ground truth data.

4. Model Architecture

I utilized the Masked Autoencoder architecture as proposed in [7]. The model was pretrained on image reconstruction with a combination of the real and synthetic data, about 4000 and 30,000 samples respectively. All of the data augmentations mentioned previously were employed, including randomized background and "leg masks" for the synthetic data. The model was pretrained for 800 epochs. See Figure 2. After pretraining, the decoder was removed and replaced with three MLP heads, one for distance, elevation, and azimuth estimation. This model, the pretrained encoder and three new heads, was then trained on the synthetic images with ground truth data for 500 epochs. I used mean squared error loss for all three labels, treating the three output logits as continuous values. Using MSE loss as opposed to

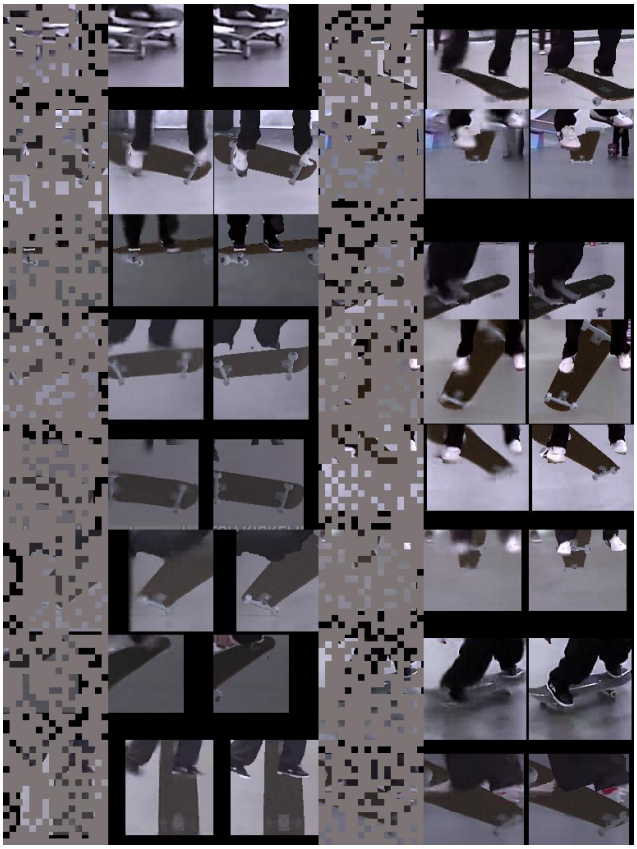

Figure 2. MAE output after pretraining on real and synthetic data. First and fourth columns are masked images. Second and fifth are the reconstructed images. Third and sixth are the original images. The image in the top left and the image in the second to last row on the right are real images, the rest are synthetic with the "leg mask", color jitter, and translation augmentations.

Augmentations on Training Set	Test Loss
None	13.155
Color and Translation	13.292
Leg Masks, Color, and Translation	11.887

Table 1. Results.

Figure 3. Example of rotation estimation on real world frame.

Figure 4. Another example of rotation estimation on real world frame.

cross entropy loss with distinct buckets meant that model predictions closer to the ground truth were rewarded and predictions farther from the truth were punished.

5. Results

As there is currently no labelled real world data to act as a benchmark for this task, I created a test set of 200 synthetic images, with the same process as described above, along with the ground truth camera angle data. I performed an ablation study with 3 different levels of data augmentation on the training set to see if it succeeded in preventing over fitting. In each of these tests all else but the training set remained constant, such as the model architecture, learning rate, and batch size. In Table 1 you can see that the best performing model is the one trained with all three data augmentations, "leg masks", random color jitter, and random translation. However, the training run with no augmentation slightly outperformed the run with just random color jitter and random translation. This suggests that the increase in performance in the training run with all three augmentations is due solely to the "leg mask" augmentation, and the two other augmentations may actually have a negative effect on performance.

It should be noted that performance on the synthetic test set should not be equated to performance on real data (see Figures 3 and 4). There remains a domain gap between the real and synthetic images, for example the skateboards in the real images have various graphics on the bottom of the board, while the skateboard 3D model being used in the synthetic data has only one texture. Clearly there is still work to be done in reducing this gap.

6. Conclusion

I propose both a new task, skateboard rotation estimation, and a new approach to said task, using a MAE with synthetic data. Accurately predicting the rotational orientation

of a skateboard is crucial for understanding and classifying skateboarding tricks. Since there is a lack of large-scale annotated datasets specifically designed for skateboard poses, I utilize a synthetic data pipeline that involves rendering 3D skateboard models with randomized camera poses and compositing them onto real-world images. The core of the proposed approach is a MAE model, pretrained on both real and synthetic data. Domain adaptation techniques are employed to bridge the gap between synthetic and real images and enhance the model's performance on real data. The results show promise, but further evaluation on real-world data are needed to fully validate the approach.

7. Future Directions

There are plenty of directions one could go in building on this work. For one, the MAE only did frame by frame analysis, despite the fact that the nature of video is temporal. Incorporating this temporal information into the model may improve prediction accuracy and stability. To do this may require generating synthetic videos rather than images.

One could also take an entirely different approach to the synthetic data generation, like using skateboard video games as simulators. Tools like Unity Perception may be helpful here.

There's also the possibility of just hand labelling a small real world dataset for testing/training. Especially for testing, this would provide a more realistic benchmark to assess model performance.

References

[1] Muhammad Amirul Abdullah, Ayan Ibrahim, Muhammad Shapiee, Muhammad Zakaria, Azraai Razman, Rabiu Musa, Noor Azuan Abu Osman, and Anwar P P Abdul Majeed. The classification of skateboarding tricks via transfer learning pipelines. *PeerJ Computer Science*, 7:e680, 2021. 2

[2] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *CoRR*, abs/2005.12872, 2020. 2

[3] Hanxiao Chen. Skateboardai: The coolest video action recognition for skateboarding, 2023. 2

[4] Bowen Cheng, Alexander G. Schwing, and Alexander Kirillov. Per-pixel classification is not all you need for semantic segmentation, 2021. 2

[5] Stephen Gibson and Clark Fitzgerald. Skateboardml: Classifying skateboarding tricks. https://github.com/LightningDrop/SkateboardML, 2021. 2

[6] Fabian Hammes, Alexander Hagg, Alexander Asteroth, and Daniel Link. Artificial intelligence in elite sports—a narrative review of success stories and challenges. *Frontiers in Sports and Active Living*, 4, 2022. 1

[7] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners, 2021. 1, 2

[8] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. Microsoft coco: Common objects in context, 2014. 2

[9] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 2

[10] Zhe Wang, Petar Veličković, Daniel Hennes, Nenad Tomašev, Laurel Prince, Michael Kaisers, Yoram Bachrach, Romuald Elie, Li Kevin Wenliang, Federico Piccinini, William Spearman, Ian Graham, Jerome Connor, Yi Yang, Adrià Recasens, Mina Khan, Nathalie Beauguerlange, Pablo Sprechmann, Pol Moreno, Nicolas Heess, Michael Bowling, Demis Hassabis, and Karl Tuyls. Tacticai: an ai assistant for football tactics. *Nature Communications*, 15(1), 2024. 1