Received: 30 August 2023 | Revised: 29 October 2023 | Accepted: 21 November 2023 | Published online: 28 December 2023

RESEARCH ARTICLE

Artificial Intelligence and Applications
2024, Vol. 00(00) 1–8
DOI: 10.47852/bonviewAIA32021615

BON VIEW PUBLISHING

Using Synthetic Data for Wheelchair Footrests Design Customization: The Kyklos 4.0 Approach

Patrick Lowin[1], Yacine Rebahi[1,*], Benjamin Hilliger[1] and Bowen Zheng[1]

[1]*Fraunhofer Fokus, Germany*

Abstract: Wheelchairs are complex systems often requiring a wide range of adjustments to adapt to the various types of patients' disabilities. They also include a series of additional elements, for example footrests, designed to keep the patient in a comfortable position. Unfortunately, most of the commercial products do not allow maintaining the position of a patient foot who has no control over his lower limbs. To address this issue, customization seems to be the appropriate solution as it enables to tailor products based on predetermined features. In Rebahi et al. [1], we have explored the use of computer vision and artificial intelligence to correctly define customized parameters of the wheelchairs' footrests. The proposed solution is based on estimating geometric properties of real shoes contours. Although this solution was accurate to some extent, its main drawback was the small amount of data that we were able to collect. For this reason, we decided to explore another approach where shoes contours data are synthetic, and convolutional neural networks (CNNs) are applied. The CNN shows promising results with minor inaccuracies of 0.8 cm created by our preprocessing. This paper discusses the synthetic data approach and compares its performance to the one described in Rebahi et al. [1].

Keywords: deep learning, design customization, wheelchair, footrest, synthetic data, Kyklos 4.0, circular economy

1. Introduction

Wheelchairs are systems helping persons with reduced mobility to become more independent and actively participate in society. According to straits research [2], the global wheelchair market size that was around 3,339 million US$ in 2022 will reach almost 5,562 million US$ in 2031. This increase is mainly driven by the aging of the population and the higher occurrence of disabilities. This report also mentions that the integration of Internet of Things (IoT), smart technologies, and customization features can be good opportunities for the wheelchair market. In fact, and according to the European Commission [3], technologies such as artificial intelligence (AI), 3D printing, and IoT are already being applied to the wheelchair industry. Data Bridge Market Research [4] also claims that the smart wheelchair market, which was US$ 150.8 million in 2021, would increase to US$ 285.38 million by 2029 and is expected to undergo a compound annual growth rate (CAGR) of 8.3% during the forecast period 2022 to 2029.

Standard wheelchairs are often prescribed for persons affected by temporary injuries or disabilities. If the disability is permanent, like paralysis, these wheelchairs are not adequate anymore and need to be customized in order to take precisely into account the patient's body measurements. For this reason, we see that more and more companies (like SORG, Sunrise Medical, and Aidacare) producing wheelchairs tend to add customization features to their products. In our work [1], the use of computer vision and AI to correctly define customized anthropometric parameters of the wheelchairs' footrests was explored. The proposed solution used parameters extracted from real shoes contours. Although this approach was accurate to some extent, its main drawback was the small amount of data that we were able to collect. In this paper, we would like to investigate another approach where shoes contours data are synthetic, and convolutional neural networks (CNNs) are applied. We will also compare the related performance results to the ones obtained from the geometric approach.

This paper is organized as follows. Section 1 introduces the topic being investigated. Section 2 discusses the related state of the art. Section 3 highlights the solution that was implemented, Section 4 presents the experimental results, and Section 5 concludes the paper.

*Corresponding author: Yacine Rebahi, Fraunhofer Fokus, Germany. Email: yacine.rebahi@fokus.fraunhofer.de

2. Literature Review

To effectively train machine learning (ML) algorithms, big datasets are needed. The larger and more diverse the datasets are, the better the model performance will be. Unfortunately, collecting real-world data and labeling them is time consuming and often difficult. Moreover, there are always issues related to privacy, copyrights, and ethics that need to be aware of [5], when collecting such data. An alternative that is gaining more and more acceptance within the ML community is the use of synthetic data to train the ML models [6]. Synthetic data are information that is artificially generated for training and testing purposes. Regardless the performance of this new approach, some of the problems related to data protection, mentioned earlier, will certainly be eliminated. Another crucial benefit of utilizing synthetic data is the fact that no manual labeling of this data is required [6].

In He et al. [7], deep learning layers were reformulated based on residual functions with reference to the layer inputs. The authors claim that these residential networks are easier to optimize and more accurate. Their experiments show that an ensemble of the residual nets achieves 3.57% error on the ImageNet test set.

In Tobin et al. [8], a technique, called domain randomization, for training models on simulated images of simple objects that transfer to real images by randomization was explored. It was demonstrated that an object detector trained only in simulation can achieve high enough accuracy in the real world to perform grasping in clutter.

In Ren and Lee [9], an additional discriminator was used in a self-supervised manner to actively minimize the domain gap learned by the model. Following a generative adversarial networks approach, one model learns to predict depth, edges, and surface normal while another discriminates between the learned features from synthetic and real data.

Similarly, Borrego et al. [10] used this technique to train a single shot detector object detector on synthetic data. They generated non-photorealistic data to fine tune the model and showed that this can substantially improve the accuracy (up to 25%) of a CNN.

In Borrego et al. [10], the authors introduced a new Syn2Real benchmark for unsupervised domain adaptation. The benchmark aims to evaluate the object recognition accuracy of models trained on synthetic data when applied to a real target domain.

In Hoffman et al. [11], a novel discriminatively trained cycle-consistent adversarial domain adaptation model was discussed. This technique, called CyCADA, is able to adapt representations at both the pixel-level and feature-level, enforcing cycle-consistency while leveraging a task loss. Moreover, CyCADA does not require aligned pairs. The authors claim that CyCADA was applied to a variety of visual recognition and prediction settings, including digital classification and semantic segmentation of roads scenes, and has shown effectiveness even on challenging synthetic-to-real tasks.

In Duan et al. [12], an approach named CenterNet, dealing with the lack of additional looks into the cropped regions in keypoint-based techniques, was investigated. Instead of a pair of keypoints, each object is detected as a triplet. The authors claim that CenterNet improves both precision and recall.

In order to build efficient models, large labeled datasets are crucial in the training phase. For this reason, the authors of Meta-Sim: Learning to Generate Synthetic Datasets [13] have developed a technique, called Meta-Sim, able to learn a generative model of synthetic scenes, and obtain images as well as its corresponding groundtruth via a graphics engine. The performed experiments on downstream tasks have shown that Meta-Sim significantly improves content generation quality over a human-engineered probabilistic scene grammar, both qualitatively and quantitatively.

In Law and Deng [14], a new approach for object detection was discussed. This approach, called CornerNet, detects an object bounding box as a pair of keypoints, the top-left corner and the bottom-right corner, using a single convolution neural network. Doing so, the need for designing a set of anchor boxes commonly used in prior single-stage detectors is eliminated. In addition to that, a new type of pooling layer that helps the network better localize corners was introduced. The authors also claim that CornerNet achieves a 42.2% AP on MS COCO, outperforming all existing one-stage detectors.

In Behl et al. [15], an algorithm for optimally generating synthetic data based on a novel differentiable approximation of the objective was proposed. The undertaken approach aims at addressing the issues of the recent methods focusing on adjusting simulator parameters usually relying on REINFORCE like gradient estimators. The authors claim that their method is faster (up to 50 ×) in finding data distribution. They also claim that the training data generation was reduced up to 30 ×, with a better accuracy (+8.7%) on real-world test datasets than the other methods.

In Hwang et al. [16], the problem of vision-based action recognition of elders' daily activities using deep learning is discussed. Based on modern visualization techniques, the authors have developed an action simulation platform, called ElderSim, that can generate synthetic data on elders' daily activities. This platform was used to generate a large-scale synthetic dataset of elders' activities of daily living, named KIST SynADL, and combined with real datasets to train three state-of-the-art human action recognition models. The authors claim that some performance improvement was noticed from the experiments undertaken on newly proposed scenarios.

In Mikami et al. [17], the authors investigated in which cases an increase in synthetic data helped to bridge the domain gap. They note no increase in data allows to bridge the gap if it is too large. They introduced a simple scaling law that predicts the performance from the amount of pre-training data.

In Kim et al. [18], a new action recognition benchmark, called SynAPT, was introduced in order to mitigate the issues related to training models with real videos, such as privacy, bias, and ethics. The authors have constructed a synthetic dataset from three publicly available assets (ElderSim, SURREACT, PHAV), trained models on the produced dataset, and then transferred these pre-trained models to various downstream tasks. The authors claim that the models pre-trained on the synthetic dataset outperform those pre-trained on real videos on the downstream datasets with low representation bias.

In Mishra et al. [19], a unified model, called Task2 Sim, that learns to map downstream task representations to optimal simulation parameters for synthetic pre-training data for them was discussed. It was shown that Task2Sim can be trained on a set of "seen" tasks and can then generalize to novel "unseen" tasks predicting parameters for them in one shot. The authors also claim that Task2Sim can compete with pre-training on real images from ImageNet.

Table 1 summarizes the analysis of the selected literature papers.

3. Methodology

In Rebahi et al. [1], we have proposed a technique (called along this paper, geometric approach) combining computer vision and deep learning to extract the dimensions of the contours of the shoe on an A4 sheet of paper. The application of a ML component is to train a model that was able to rectify the computer vision biases as opposed to the dimensions measured by hand. Although this footrest design approach was accurate, its main drawback was the small amount of data that we were able to collect. As a result, we decided to explore another approach based on CNNs and where the data are synthetic.

In this paper, the second technique (called synthetic data-based approach along this paper) will be discussed, and its results will be compared to the results of the first one. Before going deeper and for clarity sake, a short overview of the geometric approach will be given first. For more details, we refer to Rebahi et al. [1].

Table 1
Meta-level analysis of the selected relevant literature

No.	Reference	Year	Technique used	Dataset used/produced
1	Rebahi et al. [1]	2023	Combination of deep learning and computer vision	Real data based on shoes contours
2	Kim et al. [18]	2022	Synthetic data generation	Synthetic dataset produced from three publicly available assets (ElderSim, SURREACT, PHAV)
3	Mishra et al. [19]	2022	Transfer learning, deep learning	Synthetic data, model tested on 20 classification tasks
4	Hwang et al. [16]	2021	Real and synthetic data fusion for training SOTA models	A large-scale synthetic dataset of elders' daily activities, called KIST SynADL, was produced. KIST SynADL was used in addition to some real datasets
5	Ren and Lee [9]	2018	Domain adaptation, adversarial learning	PASCAL VOC 2017 classification and 2012 detection
6	Borrego et al. [10]	2018	Domain adaptation, domain randomization	Non-photorealistic synthetic data
7	Borrego et al. [10]	2018	Domain adaptation	ShapeNetCore, Microsoft COCO
8	Mikami et al. [17]	2021	Transfer learning, deep learning	More a theoretical analysis
9	Behl et al. [15]	2020	Technique based on differentiable approximation for generating synthetic data	Photorealistic renderer was used
10	Kar et al. [13]	2019	Neural networks, autoencoder loss	KITTI, ImageNet
11	Hoffman et al. [11]	2018	Cycle-consistent adversarial domain adaptation	MNIST, SVHN, SYNTHIA, GTA, CityScapes
12	Tobin et al. [8]	2017	Domain randomization technique for training models	Simulated images of simple objects that transfer to real images
13	Law and Deng [14]	2020	Convolution neural network	Microsoft COCO dataset
14	Duan et al. [12]	2019	Convolution neural network	Microsoft COCO dataset
15	He et al. [7]	2015	Residual neural networks	ILSVRC, CIFAR-10, Microsoft COCO

3.1. Geometric Approach

As described in Rebahi et al. [1], the geometric approach works as follows:

1. Contours of a patients' shoe are drawn on an A4 sheet of paper and photographed.
2. The model extracts the dimensions of an object using OpenCV. The latter is a computer vision library with which we can extract the dimensions of an object from an image if a reference object with known dimensions exists in the image. The dimensions that will be extracted by OpenCV are shoe length, top shoe width, and bottom shoe width
3. It uses the dimensions found by OpenCV and feeds them into a ML model, which predicts the anthropometric measurements of the patient's shoe.

Figures 1 and 2 give an overview of the overall structure of the implementation.

3.2. Synthetic data-based approach for shoe size estimation

CNNs and other deep learning approaches have achieved impressive results for tasks like image recognition or object detection. Large amounts of data are required to train these models. For specific tasks like object detection or segmentation, there exist annotated datasets, for example, Microsoft COCO. Creating a dataset is often tedious and labor intensive as a human has to manually annotate all images with its corresponding ground truth. Since we are the first ones to estimate shoe dimensions from RGB images, we had to create our own dataset. We take an existing dataset of footprints, calculate their outlines, and annotate them. Since the number of footprints in the dataset is small, we create synthetic data from the real data. The dataset consists of shoe outlines with the corresponding keypoints for estimating foot length, top width, and bottom width. We use these keypoints to calculate the desired anthropometric measurements. We, then, use a state-of-the-art ResNet [20] model to predict these keypoints.

3.3. Keypoint-based approaches

Estimating length from just an RGB image is not an easy task to learn for neural networks due to ambiguities created by depth and viewing direction. For example, shoes in an image appear smaller the further away they are from the camera or get distorted because of a different perspective. We take the prior knowledge of the known DIN A4 paper into account to adjust and rectify the image for a consistent input for our model. Furthermore, we simplify the approach even further and ask the network to output a belief distribution over the pixels where it expects the top, bottom, top left, top right, bottom left, and bottom right of the shoe. We then take the most likely pixel from each of these belief distributions and calculate the pixel distance ourselves. Combined with the knowledge about the DIN A4 size and its appearance in the image, we convert pixel distance to centimeters. There are several ways of representing belief distributions, and keypoint-based approaches like CornerNet [14] have shown great results in object detection, by detecting the corners of objects in an image, and are robust against inconsistencies like occlusions or in our case against non-optimal shoe outlines. We adapt this approach to our own architecture.

For our shoes, we select six keypoints: two for the top and bottom, two for the top left and top right, and two for the bottom left and bottom right of the shoe. Using the six keypoints, we can calculate the length, top width, and bottom width.

Figure 1
Footrest design customization implementation. This figure shows how computer vision and machine learning are combined to extract the anthropometric measurements

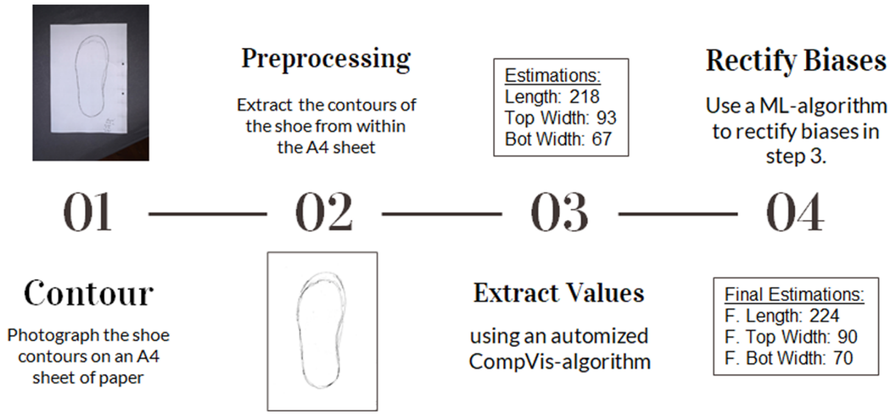


Figure 2
Principle dimensions and measurements are needed for the design of the footrest

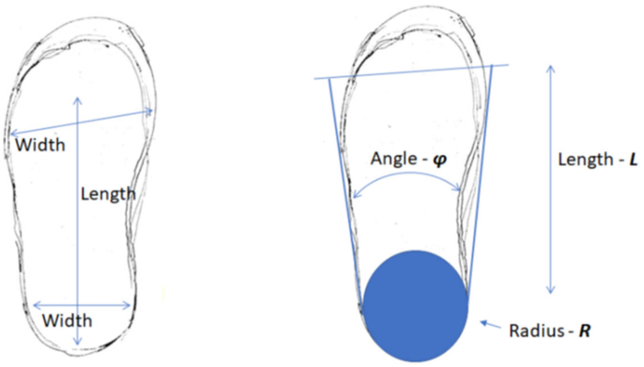
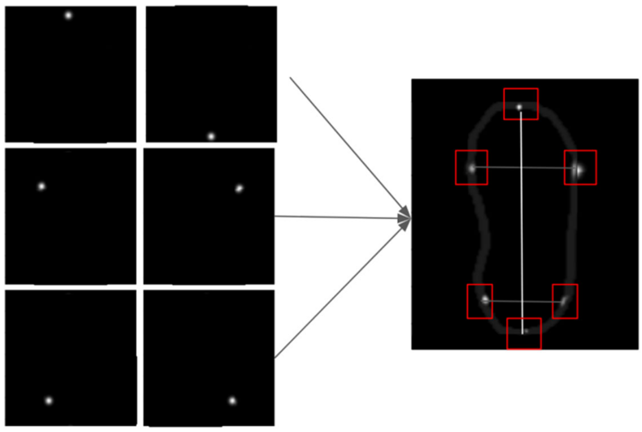


Figure 3
Detecting keypoints on a shoe contour



3.4. Dataset creation and data alterations

To train a neural network, we decided to create our own shoe size estimation dataset, where shoe outlines lines are labeled with keypoints for length, top and bottom width as can be seen in Figure 3. For this, we use the Footwear Impression Database [21], which provides high-quality scans of footprints. Our approach is based on the work in domain randomization [8], which showed that neural networks are able to bridge the gap between synthetic and real-world data given enough variation inside the synthetic training data. For this, we perform alterations to the shoe outlines generated from the Footwear Database in the form of scaling and deformations to create our synthetic dataset.

Using basic image processing techniques, we first create a binarized version of our image using Otsu's method [22] to find a suitable threshold. We apply the morphological operations opening and closing to the footprint to close gaps and create shapes with extractable contours, and we, therefore, apply dilations and closing to close small gaps. After that, we use OpenCV to extract the contour of our processed image. Since keypoints can vary between shoes, we had to annotate them by ourselves. For this, we wrote a simple annotation tool, where the user draws three lines indicating where the length, top width, and bottom width should be calculated. Since the user is not always clicking on the outline, we select the point on the shoe closest to the end of the drawn line as our keypoint. Because we do not know the actual size of the footprints, we resize all images to 297 × 210 pixel where each pixel corresponds to a millimeter. This allows us to resize the shoes to arbitrary shoe sizes.

It is worth to mention that we have created shoe contours from a footprint database and annotate them with ground truth keypoints. Then we apply augmentations to create our synthetic dataset.

The dataset is restricted to only 1175 images, which is not enough to train a CNN. To create more variety, we performed alterations to our data which are illustrated in Figure 4. We calculate three points in our shoe and place Gaussian bells on them. This creates a deformation field which allows us to create more shoe shapes. For the final input to our network, we also rotate and flip to prevent overfitting. In Table 2, the terminology related to the used dataset is provided.

3.5. Network architecture and training

For our CNN architecture, we use a ResNet-18, seen in Figure 5 and replaced the last layers with our heatmap module which consists of three convolution blocks where we reduce the number of features in each step. The final output of each block is a 128 × 128 heatmap. To train our architecture, we use a variant of focal loss first

Figure 4
Data creation pipeline

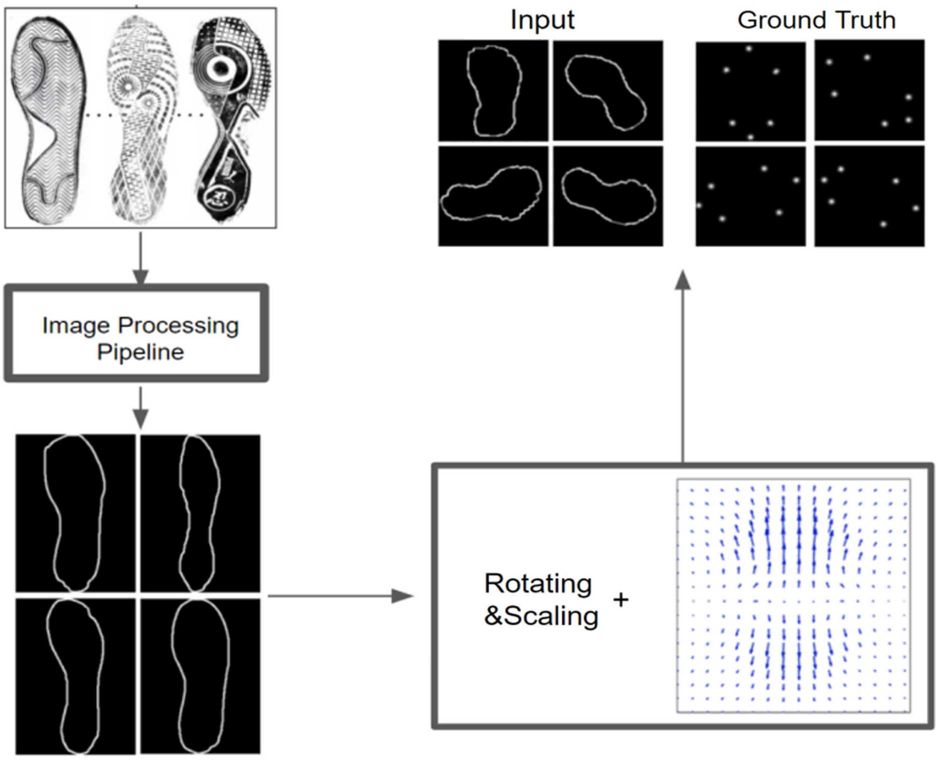


Table 2
Dataset dictionary

Technique used	Explanation
Outline detection	Binarization of the footprint, contour extraction using OpenCV
Manual annotation	Annotation of the outline with 6 keypoints
Deformation field	Deforms the outline at the top and bottom, e.g., smaller or wider shoe outlines
Rotation and scaling	Rotating and scaling the outline to create different shoe sizes
Heatmap	Placing a 2D Gaussian over the keypoints

introduced by CornerNet [14] and as our optimizer we use Adam with a learning rate of 0.0001. For training, we do not need to resize the images back to their original size of 297 × 210 because focal loss and the keypoints are independent of the actual length. We trained our architecture on the P100 GPU provided by Kaggle [23] to speed up the training. To evaluate and use the network in a real-life task, we need to resize the heatmap to the original image size. We test our trained network on 24 shoe outlines which we excluded from the training data to evaluate our approach.

It is important to mention that we have used a ResNet architecture as our backbone and replaced the fully connected layer with a heatmap module consisting of multiple convolution layers to output heatmaps.

4. Evaluation

We evaluate our network on the real-life foot sketches. We convert the foot sketches to a binary contour using the computer vision algorithm described earlier and feed them to our network. The predictions can be seen in Figures 6 and 7 and show promising results and correctly detected keypoints even on the real-life data.

On the right side, we see the original image, which is the input for the computer vision algorithm, on the left side, the binarized output, and in the middle the aggregated heatmaps over the shoe outline. We cut out the shoe and made it bigger to improve the visibility of the activity blobs. As one can see, the network correctly predicts the points on the outline where we measure the distances. One downside of the keypoints is that we also measure diagonally which increases the estimated length or width. This can be improved by looking for the nearest point on the outline in y-direction for the height and in x-direction for the width. We evaluate our deep learning approach on the real-world data. We calculate the mean and standard deviation for the errors occurred in this approach.

	Mean error	Standard deviation
Shoe length	8.68 mm	5.52 mm
Top length	7.77 mm	9.18 mm
Bottom width	5.54 mm	4.50 mm

To transform our keypoint predictions from pixel to metric space, we perform scaling to the measured distances with the size of a DIN A4 paper divided by the extracted binary image size from the real-world approach. Since the viewing angle is not always perfectly perpendicular to the paper, we encounter some minor scaling errors that are mainly responsible for the errors listed above.

To compare the algorithmic to the deep learning approach, we compute the mean and median error for all estimations.

Figure 5
Modified ResNet architecture

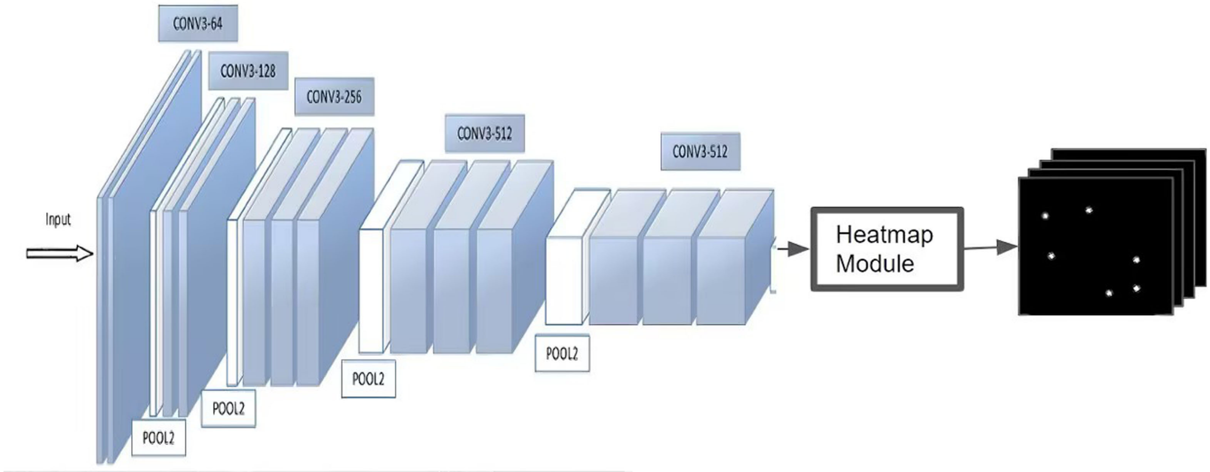


Figure 6
Predicted keypoints on real data

Figure 7
Outlier for the geometric approach

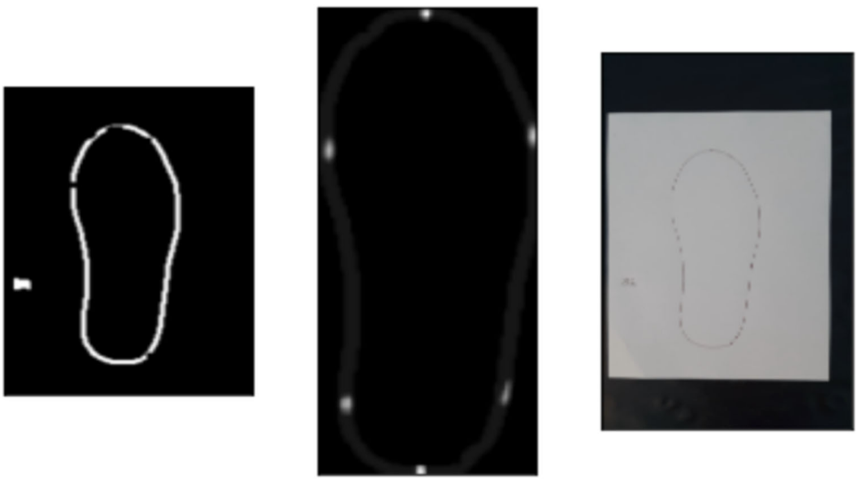
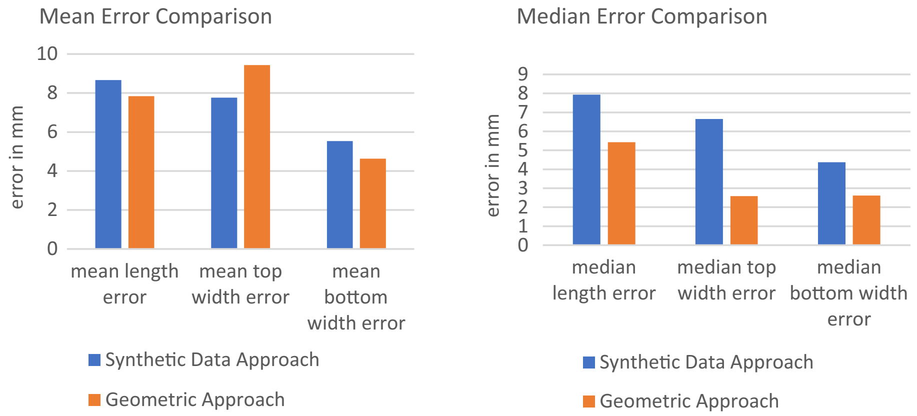


Figure 8
Comparing the synthetic data approach and the geometric approach



In Figure 8, we plot the mean and median error for our real and synthetic data approaches. We observe that geometric estimation using triangles and circles is more accurate as seen in the median errors. However, the mean and median errors for the synthetic data approach are closer, meaning that we have fewer outliers with large errors. The following image shows an example. Here the line is not connected, and the algorithm fails to estimate bottom width. The main advantage of the synthetic data approach is the ability to generalize to a large variety of shoes with acceptable accuracy. However, for estimating small shoes, our analysis shows that the geometric approach is to be preferred.

5. Conclusion

In this paper, a new technique, called synthetic data based approach, for supporting the design of wheelchairs footrests was explored. It is based on the use of deep learning networks trained on synthetic data. In fact, this technique was proposed to deal with the lack of sufficient shoes contours data that were utilized in the development of the [1] solution. The latter is also a technique, mainly based on computer vision, for extracting anthropometric measurements needed in the design of the wheelchairs footrests. The technique discussed in this paper adapted keypoint-based approaches such as CornerNet that have shown satisfactory results in object detection and robustness against inconsistencies like occlusions, namely non-optimal shoe outlines in the context of this paper. This technique was implemented and tested, and the obtained results have shown that its main advantage is the ability to generalize to a large variety of shoes with acceptable accuracy. However, for estimating small shoes, our analysis shows that the geometric approach elaborated in [1] is to be preferred.

Funding Support

This work was carried out within the H2020 KYKLOS 4.0 project (Grant Agreement Number 872570)[1]. This project, funded by the European Commission, life cycle and promoting energy efficient and low material consumption, aims at developing an innovative circular manufacturing ecosystem based on novel circular production systems and artificial intelligence technologies, targeting personalized products with extended life cycle and prom.

[1]https://kyklos40project.eu/vision/

Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.

Data Availability Statement

Data available on request from the corresponding author upon reasonable request.

References

[1] Rebahi, Y., Gharra, M., Rizzi, L., & Zournatzis, I. (2023). Combining computer vision, artificial intelligence and 3D printing in wheelchair design customization: The Kyklos 4.0 approach. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA32021275

[2] Straits Research. (n.d.). *Wheelchair market: Information by product (Manual wheelchair, power assisted manual wheelchair), category (adult, pediatric), applications (homecare, hospitals, ambulatory surgical centers, rehabilitation centers), and region – Forecast till 2031*. Retrieved from: https://ec.europa.eu/research-and-innovation/en/horizon-magazine/smart-tech-gives-new-spin-classic-wheelchair

[3] Jensen, C. (2021). *Smart tech gives a new spin on the classic wheelchair*. Retrieved from: https://projects.research-and-innovation.ec.europa.eu/en/horizon-magazine/smart-tech-gives-new-spin-classic-wheelchair

[4] Data Bridge Market Research. (2022). *Global smart wheelchair market – Industry trends and forecast to 2029*. Retrieved from: https://www.databridgemarketresearch.com/reports/global-smart-wheelchair-market

[5] Zewe, A. (2022). *In machine learning, synthetic data can offer real performance improvements*. Retrieved from: https://news.mit.edu/2022/synthetic-data-ai-improvements-1103

[6] Datagen. (n.d.). *Synthetic data: The complete guide*. Retrieved from: https://datagen.tech/guides/synthetic-data/synthetic-data/

[7] He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. *arXiv Preprint*:*1512.03385*.

[8] Tobin, J., Fong, R., Ray, A., Schneider, J., Zaremba, W., & Abbeel, P. (2017). Domain randomization for transferring deep neural networks from simulation to the real world. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 23–30. https://doi.org/10.1109/IROS.2017.8202133

[9] Ren, Z., & Lee, Y.J. (2017). Cross-domain self-supervised multi-task feature learning using synthetic imagery. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 762–771.

[10] Borrego, J., Dehban, A., Figueiredo, R., Moreno, P., Bernardino, A., & Santos-Victor, J. (2018). Applying domain randomization to synthetic data for object category detection. *arXiv Preprint:1807.09834*.

[11] Hoffman, J., Tzeng, E., Park, T., & Zhu, J. Y. (2018). CYCADA: Cycle consistent adversarial domain adaptation. *International Conference on Machine Learning*, 1989–1998.

[12] Duan, K., Bai, S., Xie, L., Qi, H., Huang, Q., & Tian, Q. (2019). CenterNet: Keypoint triplets for object detection. *arXiv Preprint:1904.08189*.

[13] Kar, A., Prakash, A., Liu, Y., Cameracci, E., Yuan, J., Rusiniak, M., . . . , & Fidler, S. (2019). Meta-Sim: Learning to generate synthetic datasets. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4551–4560. https://doi.org/10.1109/ICCV.2019.00465

[14] Law, H., & Deng, J. (2020). CornerNet: Detecting objects as paired keypoints. *International Journal of Computer Vision*, *128*(3), 642–656.

[15] Behl, H. S., Baydin, A. G., Gal, R., Torr, P. H., & Vineet, V. (2020). Autosimulate: (quickly) learning synthetic data generation. *European Conference on Computer Vision*, 255–271.

[16] Hwang, H., Jang, C., Park, G., Cho, J., & Kim, I. J. (2021). Eldersim: A synthetic data generation platform for human action recognition in eldercare applications. *IEEE Access*, *11*, 9279–9294. https://doi.org/10.1109/ACCESS.2021.3051842

[17] Mikami, H., Fukumizu, K., Murai, S., Suzuki, S., Kikuchi, Y., Suzuki, T., . . . , & Hayashi, K. (2021). A scaling law for synthetic-to-real transfer: How much is your pre-training effective? *arXiv Preprint:2108.11018*.

[18] Kim, Y. W., Mishra, S., Jin, S., Panda, R., Kuehne, H., Karlinsky, L., . . . , & Feris, R. (2022). How transferable are video representations based on synthetic data? *36th Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

[19] Mishra, S., Panda, R., Perng Phoo, C., Chen, C. F., Karlinsky, L., Saenko, K., . . . , & Feris, R. S. (2021). Task2Sim: Towards effective pre-training and transfer from synthetic data. *arXiv e-prints: 10.48550/arXiv.2112.00054*

[20] Pytorch Team. (n.d.). *Deep residual networks pre-trained on ImageNet*. Retrieved from: https://pytorch.org/hub/pytorch_vision_resnet/

[21] Kortylewski, A., Albrecht, T., & Vetter, T. (2015). Unsupervised footwear impression analysis and retrieval from crime scene data. *Computer Vision-ACCV 2014 Workshops: Singapore*, 644–658.

[22] Murzova, A., & Seth, A. (2020). *Otsu's thresholding with openCV*. Retrieved from: https://learnopencv.com/otsu-thresholding-with-opencv/

[23] Kaggle. (n.d.). *P100 GPU*. Retrieved from: https://www.kaggle.com/code/raimondomelis/preprocessing-in-chunks-with-gpu-p100-vs-t4-x2

How to Cite: Lowin, P., Rebahi, Y., Hilliger, B., & Zheng, B. (2024). Using Synthetic Data for Wheelchair Footrests Design Customization: The Kyklos 4.0 Approach. *Artificial Intelligence and Applications*. https://doi.org/10.47852/bonviewAIA32021615