Machine Learning and Intelligent System

Project Update Report
POSE ESTIMATION OF SNEAKERS IN A PICTURE

by
Eugène Berta
Patrick Iversenc

Submission Date: December 7, 2020
Supervisor: Maria Zuluaga

1 Introduction

1.1 Motivation

This student project deals with the issue of pose estimation. It is a widespread problem in the fields of image processing and deep learning. We decided to limit ourselves to the specific case of sneaker's photographs. From a picture containing a sneaker, our goal is to extract a 3D model and positioning of the shoe in the repository induced by the camera.

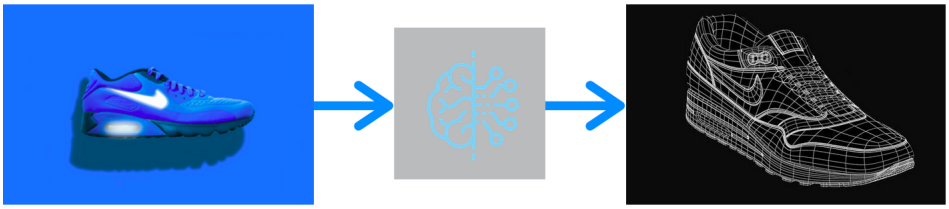

Figure 1: Input to output, motivation schema

1.2 Applications

Pose estimation has a variety of applications, especially in image manipulation. In the case of sneakers, estimating the 3D position of a shoe in a picture is critical if you want to replace a shoe in a picture, to change its colors or shape for example. Such applications can be used in advertising, or to help designers and creators in the fashion industry. Developing these technologies is one of the goals of the french startup "Smartpixels", specialized in digital technologies for retail in the luxury and textile industries. We have been working with this startup to develop our 3D pose estimation model, using their technologies to implement an innovative method for pose estimation.

1.3 Method

The method we decided to implement is based on two major steps :

1. The first step is "instance segmentation". It consists in extracting the pixels that are part of the sneaker in the photograph. This first step will give us the "shape" of the shoe in the image.
2. The second step is to use this shape to infer the 3D positioning of the shoe in a camera based repository.

To implement the first step, we used existing deep learning technologies, widely used for image segmentation. We will present you this in the first part of our report. For the second step, we worked with the startup Smartpixels to develop an original model, predicting the orientation of the shoe by comparing it's 2D shape with views from a sneaker's 3D model. We will present you this work in the second part of the report.

2 Sneakers segmentation and Deep Learning

In order to implement a deep learning algorithm for image segmentation, we used "Detectron2", a software system based on PyTorch, edited by Facebook AI Research. Detectron2 helps developers implement object detection algorithms. We first used Detectron2 pre-trained models to segment pictures, we then decided to train our own model, with our data set.

2.1 Building our dataset

In order to train Detectron2 models with our own data, we needed to create an adequate data set for image segmentation. The disadvantage of image segmentation is that it requires the pictures in the data set not only to be labeled based on what the picture contains, but they also need to be segmented. Such segmented images are pretty hard to find on the internet, so we decided to create our own data set. The first step was to gather a important number of sneaker images, which we were provided with by the teams at Smartpixels. These data sets were build with researches on google images so the second step consisted in cleaning the data set and keeping only the most relevant pictures. Finally, we had to label and segment the sneakers in the images. For this task, we used a Python-based software named "labelme".

After segmenting sneakers in the image, labelme provides us with a json file containing the coordinates of the points used for segmentation. We were able to convert our input files (photographs) and output files (json files) into a COCO data set, that we used to train our model.
Using this method, we labeled one hundred sneaker pictures from our first data set, creating a first training set for our deep neural network.

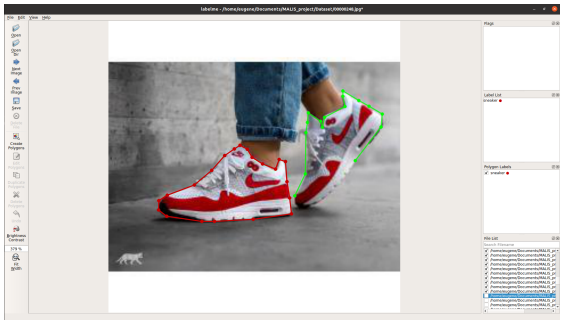

Figure 2: Labeling pictures with labelme

2.2 Second Method for generating data

As you can imagine, this first method for creating by hand a segmentation data set is very tedious and not really scalable.
After a few researches, we read about a second method, that we may implement in the second part of this project, to obtain better results with our segmentation algorithm. This method, described in the reference [1], consists in superposing contoured sneakers images with random background images. This way, knowing how we superposed the two pictures, we know where the sneaker is in the final picture, without having to segment it by hand, and we can feed those images to the neural network.

Figure 3: New method for generating data

2.3 Results

Using our first training set, composed of eighty images only (we kept twenty for testing), we trained our deep neural network with one hundred iterations and obtained pretty decent results. However, the network struggles when we tested with more complex images.

Figure 4: Simple images from the testing set, good results from our model

Figure 5: More complex images from the testing set, bad results from our model

2.4 Further work

These first results, even if they are satisfying, can be improved a lot in different ways, which lets us hope for very good results in the near future.

- First, using the second method we provided, we can hope to train our model with a much wider data set, which is likely to provide us with better results. Moreover, this method will produce images with multiples elements in the background. Our current dataset contains many pictures with sneakers only and a blurry background, this lack of other elements is not a good thing for training. As you see in our results, elements in the background (a face for example), tends to be segmented as a sneaker by our first model.
- Then, with this new dataset, we will start training our model in a more precise manner, exploiting the accuracy results to train the network with good parameters, which we didn't do for the moment.

Implementing these two improvements, we hope to reach a very good segmentation model in the near future, a model that will put us in a position to start implementing the second step of our pose estimator.

3 Pose estimation

In this second part of our report, we will expose you how we intend to estimate the pose of a sneaker in a 3D referential, and what we already did to reach that goal.

3.1 Smartpixels, silhouette generator

As we introduced before, we have the chance to learn from and work with the teams of Smartpixels, a french startup specialized in virtual imaging and retail for sportswear and luxury brands. Smartpixels is interested in the problem of pose estimation for commercial use.

When we first discussed our project with developers from Smartpixels, they exposed us many ideas that could help us in our project. The company develop photo-realistic 3D models of their clients products, and they offered us to use these models to train our algorithm.
We had to decide how to use these 3D models to best help us with our problem. Using our deep learning algorithm, we are able to extract the silhouette of the shoe in the picture. On the other hand, from a 3D model generating views of a sneaker, we can generate silhouettes of a shoe from any point of view and distance around the shoe. This is the idea we got and we decided to implement :

1. First, we use our deep neural network to extract the silhouette of a sneaker in a picture.
2. Then, we use Smartpixels' software to generate silhouette of a sneaker from different perspectives.
3. Using machine learning techniques from the MALIS course, we find the orientation and distance to the camera of the sneaker that gives us the best match between the two silhouettes.
4. We change the referential from sneaker to camera to estimate the orientation of the shoe in the input image, using the camera parameters we just obtained.

Figure 6: Pose estimation, from image to 3D Model

To help us implement this method, the developers from Smartpixels provided us with a silhouette generator, that return silhouettes from an Air Max 1 Nike sneaker. The silhouette are generated using a Unity server, and a node server running on our local machine enables us to require silhouettes for a specific orientation and position of the camera.

The parameter "fieldOfView" corresponds to the opening of the camera, we can fix it to an average value. Then, the parameters Tx, Ty and Tz correspond to the translation of the camera from the center of the shoe. Finally the parameters Rx, Ry and Rz correspond to the camera rotation on the different axis of the shoe base referential.
A simple change of referential will give us the rotations and translations in the camera referential when we will have optimized this set of six parameters.

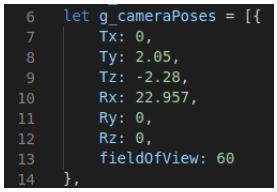

Figure 7: Parameters to set for generating a silhouette

3.2 Finding best match between silhouettes and optimizing parameters

We need to define a method to compute the distance between two binary images : our silhouette extracted from the input image and the silhouette given by the 3D model of the shoe. This distance function will be the first step into optimizing the six parameters to estimate the position of the sneaker.

Using this distance, we will use Machine Learning techniques such as gradient descent (for example) to get the best match between the two silhouettes, and so, the estimated position of the sneaker in 3D.

4 Conclusion, next steps and contributions

4.1 Conclusion and next steps

For the first part of our project, the next step consists in improving our results, following the steps we already sated. We still have a great deal of work to do in the second part of the project : figuring out how to compute the distance between the silhouettes, find an appropriate method to optimize our parameters and implement the whole.

4.2 Contributions

We have been working together for most of the project, as we live in the same house next to the school. Each one of us has participated in every step of the project : researches about neural networks and instance segmentation, implementing Detectron2, building a dataset, exchanging with Smartpixels on our method, working with the silhouette generator...

References

[1] Javier Martínez-Cesteros, Gonzalo López-Nicolás, "Automatic Image Dataset Generation for Footwear Detection", Grupo de robótica, percepción y tiempo real (RoPeRT)Instituto de Investigación en Ingeniería de Aragón (I3A) Universidad de Zaragoza, Mariano Esquillor s/n, 50018, Zaragoza, Spain. https://www.readcube.com/articles/10.26754%2Fjji-i3a.003547

[2] Gilbert Tanner, (2020). Detectron2 Train a Instance Segmentation Model. https://gilberttanner.com/blog/detectron2-train-a-instance-segmentation-model

[3] Chengwei Zhang, (2019). How to train Detectron2 with Custom COCO Datasets . https://www.dlology.com/blog/how-to-train-detectron2-with-custom-coco-datasets/

[4] Official Git Repository for Detectron2. Facebook Research. https://github.com/facebookresearch/detectron2

[5] Pascale Monasse, Kimia Nadjahi. Classez et Segmentez des Données Visuelles. https://openclassrooms.com/fr/courses/4470531-classez-et-segmentez-des-donnees-visuelles

[6] Derrick Mwiti (2020). Image Segmentation in 2020: Architectures, Losses, Datasets, and Frameworks. https://neptune.ai/blog/image-segmentation-in-2020

[7] S. Song and T. Mei, "When Multimedia Meets Fashion," in IEEE MultiMedia, vol. 25, no. 3, pp. 102-108, July-Sept. 2018, doi: 10.1109/MMUL.2018.2875860. https://ieeexplore.ieee.org/document/8589035