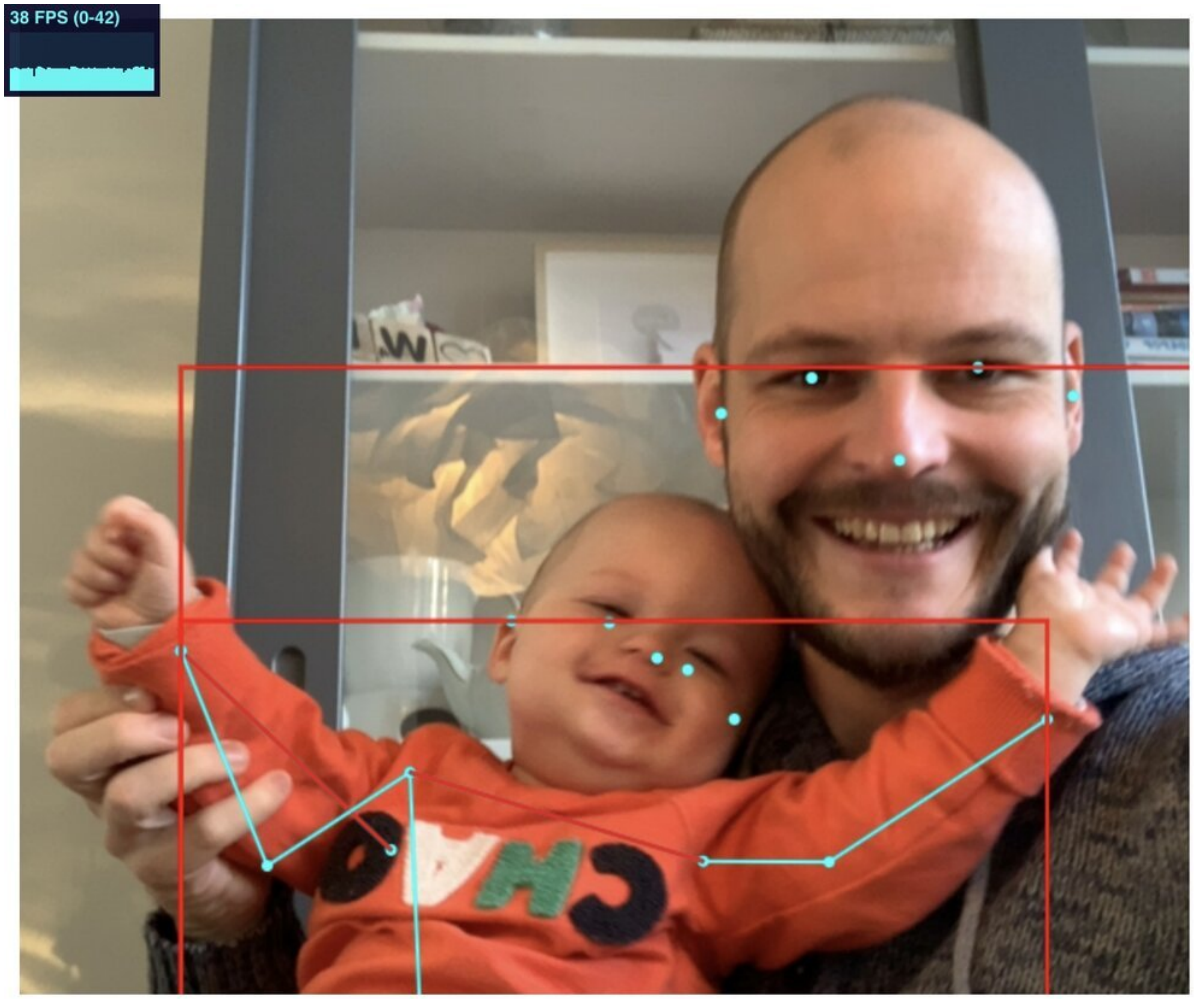
Realtime human pose recognition through computer vision

Using TensorFlow and PoseNet on a video feed

Willem L. Middelkoop
Dec. 1, 2019



For an exciting new project I have been experimenting with computer vision using TensorFlow. I wanted to achieve realtime human pose detection to drive interactive video projections and games. Time to dive into the world of machine learning, tensors and computer vision!

Computer vision

Really exciting is the interdisciplinary scientific field of computer vision, dealing with how computers can be made to gain an high-level understanding from digital images and video

feeds. It includes challenges to acquire, process and analyse digital images. Ultimately you'll want the computer to take decisions based on its understanding of the (complex) world around it.

TensorFlow

TensorFlow is an end-to-end open source platform for machine learning, originally developed by Google. It is based on scientific work from various scholars and has since become a stable and robust ecosystem of tools, libraries and community resources.

The name ”TensorFlow” is derived from the concept of a ”flow of tensors”. A tensor is best described as a ”thing”, like something that the computer recognised - or tries to analyse. Technically, a tensor is a multi-dimensional array with numeric values. This allows the computer to compare different ”things”, see how their characteristics are alike (or different).

The main challenge in realtime image processing is to reduce the complexity of the image into processable chunks that can be expressed as tensors. Usually this involves many steps comprising the computer vision's algorithm. Steps like, recognising a shape, cropping into that particular shape, resizing it, removing its colour, comparing the contours, etc. Tensors flow through these steps, from 'unanalysed' into 'recognised', henche the name TensorFlow.

Processing Power

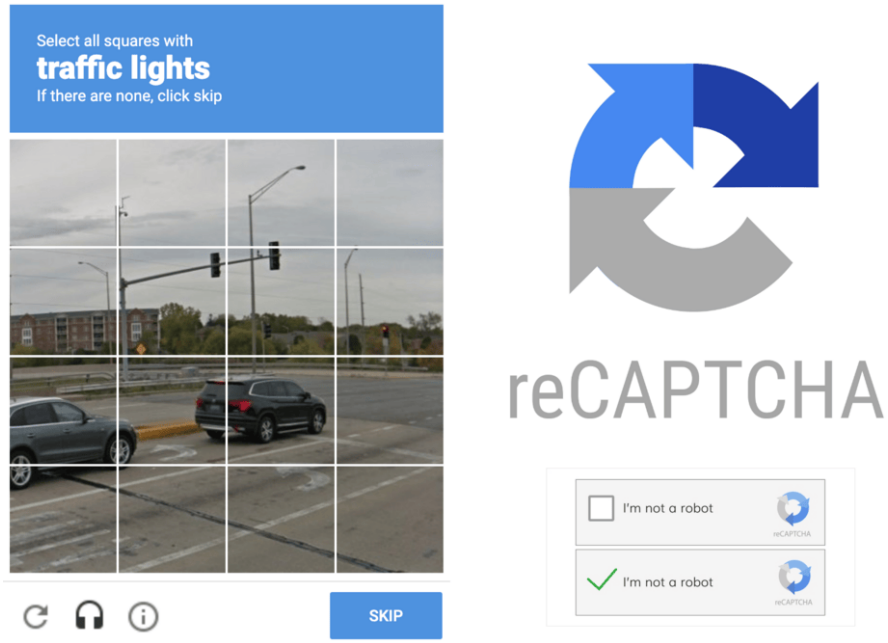
The great thing about modern machine learning using TensorFlow is that you generally do not need (very) expensive hardware to achieve results. This is because the ”heavy lifting” is done when training machine learning models, which you can sort of imagine as ”reference tables”. These models are created by analysing known datasets, where the computer is given the 'answer' and it determines the characteristics of the subject. Training models requires time, processing power and training data.

Training data

Google did something truly clever, it distributed the training of its machine learning models to all of us. Yes, that includes you!

Recaptcha

Chances are that you have encountered ”reCAPTCHA” or ”Recaptcha”. A visual challenge you have to complete when you order something online or try to register somewhere. This challenge provides a protection against (spam)bots for webshop and website owners. Its aim is to distinguish humans from bots. It does that by providing (complex) images to humans, asking them to identify objects in these images. Like traffic signs, crossings, cars, etc. Google uses this human feedback to train its models.

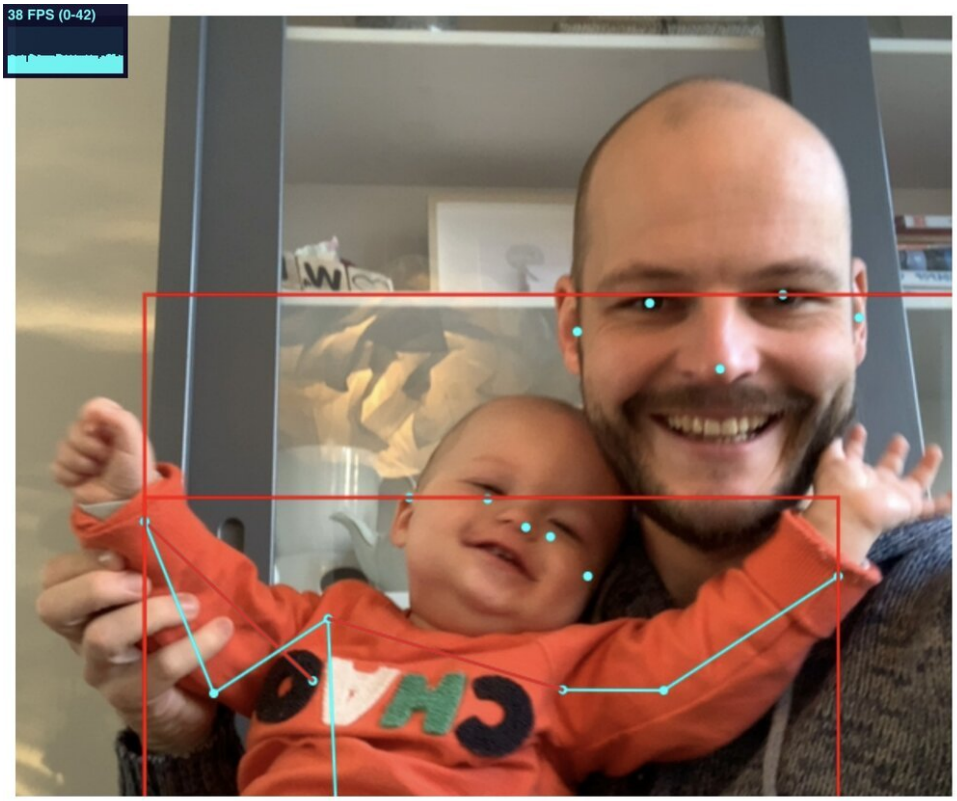


ReCaptcha: you've been training Google's AI for years!

The great thing is that many of Google's findings are made available publicly, there are many interesting open source projects that allow us - simple developers - to do amazing things with machine learning.

Human pose recognition using: PoseNet

One of these available machine learning models is PoseNet. It's a vision model that can be used to estimate the pose of a person in an image or video. This is done by determining where the key body joints are, like your elbows, hands, hips, knees ankles, etc.

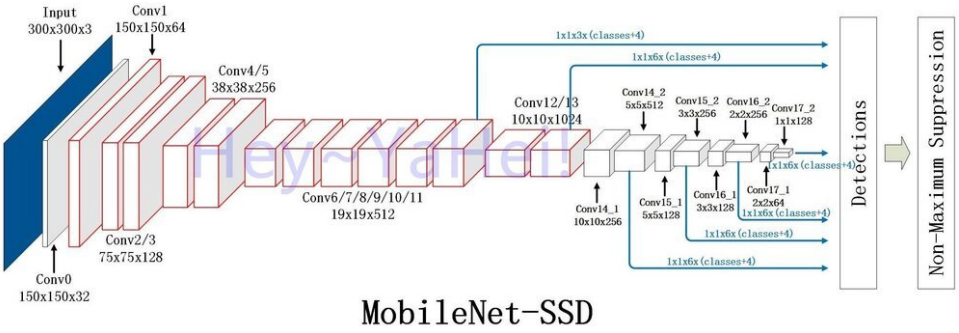


Me and mini-me being recognised by the PoseNet model

Single Shot MultiBox Detector (SSD) algorithm

The Single Shot MultiBox Detector is a popular algorithm to detect objects in images and video feeds. It is popular because of its speed. The principle is relatively simple to understand: the algorithm tries to reduce the image complexity as quickly as possible, only doing deep analyses on shrunken, simplified feature maps.

This approach works best if you want to detect large objects, as they stand out quickly: generating enough high level features to do predictions early on in the algorithm. This is great for body detection, which is one of the first steps when you want to determine human poses (first look *where* a person is, then *what* his/her pose is).



MobileNet-SSD algorithm (image: hey-yahei.cn)

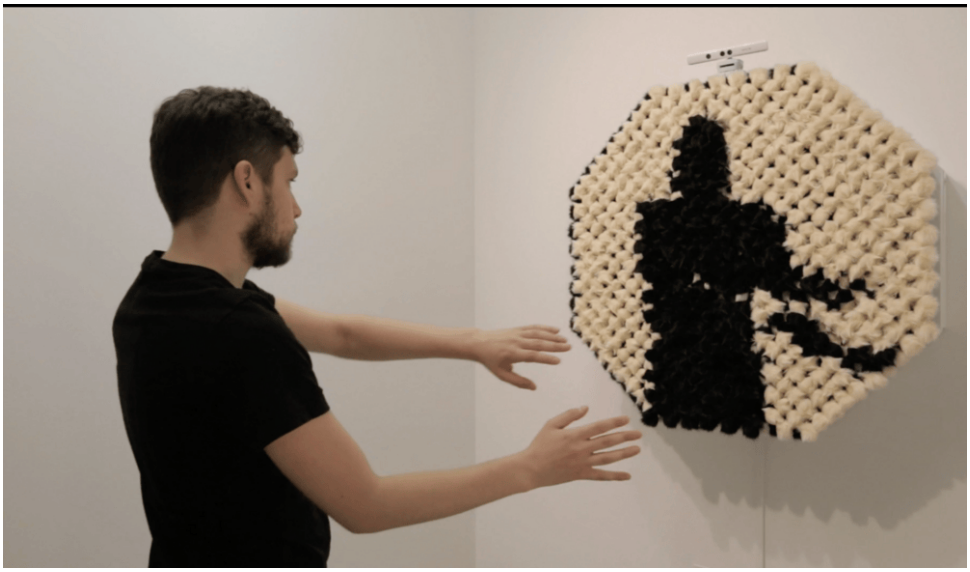
Image and person recognition is something where the people from China are pretty good at, for various reasons. Their algorithms perform well on relatively weak hardware. This enables computer vision on all kinds of devices to work well, in realtime!

Cool use cases

Being able to detect poses in realtime enables all kinds of cool stuff. You - quite literally - can become the game controller or input for an interactive art installation!

PomPom mirror

With 928 spherical faux fur puffs this art installation by Daniel Rozin is amazing. The sculpture is controlled by hundreds of motors that build silhouettes of viewers using computer vision.



PomPom faux puff mirror by Daniel Rozin

Turning you in a bird: The Treachery of Sanctuary

Enabling people to loose themselves for a moment was only possible by bringing different disciplines together. This amazing interactive art installation directed by Chris Milk is a giant triptych that takes viewers through various stages of flight using computer vision.

Turning people into birds using computer vision

Conclusion

Working on creative and experimental projects is an incredible privilege, it is what makes my work so exciting.

With ever increasing heaps of data, digital and autonomous sense-making is a major challenge for the future: one I am keen to work on!