Vision Transformers for obstacle avoidance in simulation environment

Karol Szurkowski - kaszu19@student.sdu.dk

1 Project description

Idea of the project is to apply Vision Transformer [1] network architecture so that a robot in an simulated environment can learn to avoid obstacles on a highway trained in a supervised manner as presented in the figure 1.

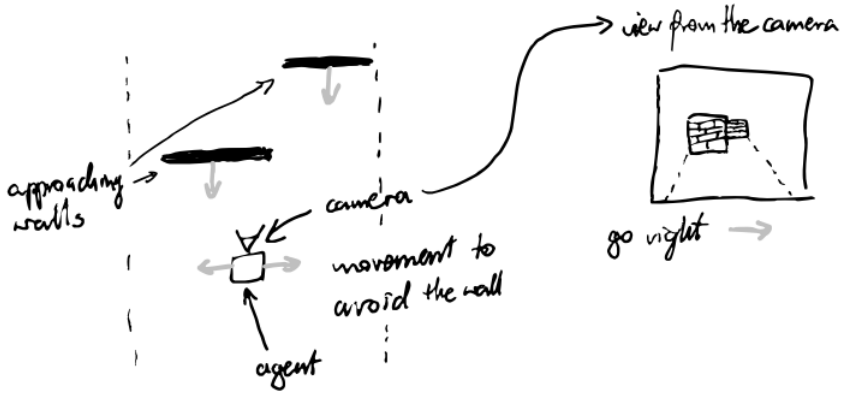

Figure 1: Simple idea presenting the simulated world and the problem to be solved.

Simulated environment will be done with PyBullet physics engine [2], which will resemble a simple game where the goal is to drive a car on a highway and avoid approaching obstacles. Setting up the simulation shouldn't be too challenging.

Training data will be generated by randomly spawning obstacles in front of the agent, which will be moving towards it. The agent will avoid the obstacles knowing their true positions from the physics engine - this is how the training data will be obtained. On each step the picture from the camera mounted in front of the robot will be saved along with the horizontal coordinates of the place where robot should move to avoid obstacle, given the turning speed and sampling time of taking a new picture.

If the regression prediction won't be achievable, the problem will be solved using classification - turn left, turn right or don't turn, which simple code was found in [3]. For what is understood now, to get the regression solving network the mlp head of the architecture needs to be changed.

The outcome of the trained network will be the steering that the object has to take to avoid the obstacle given the current state observed with an RGB camera.

The focus of the project isn't necessarily the results but the journey of understanding and applying the Transformer architecture as seen in the figure 2 to easily understandable problem, which seems more exciting than a classification.

2 Code and inspirations

Vision Transformers were first introduced in this paper [1], which explanation can be found in the video format here [4]. Working example code of using vision transformers can be found here [3] with explanations of the architecture and other code examples here [5] and [6].

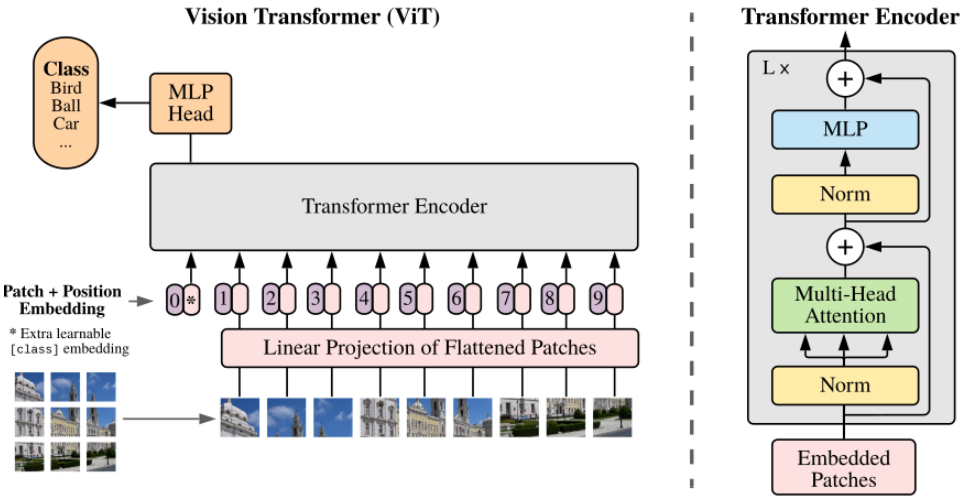

Figure 2: Vision Transformer architecture presented in [1].

References

[1] Alexey Dosovitskiy et al. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale*. 2021. arXiv: `2010.11929 [cs.CV]`.

[2] *Bullet Real-Time Physics Simulation*. URL: `https://pybullet.org/wordpress/`.

[3] Stan Kriventsov. *A Practical Demonstration of Using Vision Transformers in PyTorch: MNIST Handwritten Digit Recognition*. URL: `https://towardsdatascience.com/a-demonstration-of-using-vision-transformers-in-pytorch-mnist-handwritten-digit-recognition-407eafbc15b0`.

[4] Yannic Kilcher. *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale (Paper Explained)*. URL: `https://youtu.be/TrdevFK_am4`.

[5] *Transformer attention mechanism*. URL: `https://d2l.ai/chapter_attention-mechanisms/transformer.html`.

[6] *Vision Transformer (ViT)*. URL: https://huggingface.co/transformers/model_doc/vit.html.