

WASP Software Engineering Course Module 2024

Cuong Le

August 2024

1 Introduction

My PhD research focuses on studying human motions captured from videos. Despite being one of the most popular topics in Computer Vision research, the problem of recovering accurate 3D human motions from videos is still partially solved. This is due to the very challenging nature of the monocular camera setup, where we can only get access to 2 out of 3 space dimensions, leading to implausible motion results. My PhD research attempts to solve this by utilizing motion physics, using the noisy estimation from videos as priors. Specifically, the method is conducted inside a physics simulation, starting with a neutral simulated human body. Machine learning models and automatic control algorithms are used to compute the correct joint forces to move body parts forward, imitating the visual evidence from the videos. By approaching the problem this way, the reconstructed 3D human motions inherit the physical constraints while still respecting the visual estimation.

Having a correct motion captures and joint forces predictions can contribute to many applications, especially in sport analysis. Current state of the art technologies in, i.e. football studies, often require the players to wear intrusive sensor devices and perform the techniques within a dedicated environment. Using cameras can alleviate these constraints, allowing the players to keep playing on the field as usual, and possibly leads to better analysis since now the captured motions reflect the actual behaviors of the players in-game.

2 Relevant points from Robert's lectures

Data validation. Data errors can adversely affect the quality of the generated model. I agree with this statement since I also came across the same problem while developing my ML models, especially in the beginning stage of the research. Input human motions captured from camera can get very noisy or even corrupted when encountering view occlusion. This is often overlooked when given large datasets and can lead to large decrement in ML model performance.

Model development. Accuracy vs explainability. ML models, especially deep learning models are often considered as “black box” since it is almost impossible to interpret what types of features that the models learned to make its predictions. Without knowing exactly what a model learns and does, we can run into serious uncertainty problems when the model is so confident in its incorrect predictions. Integrating physics knowledge into a ML system as I am doing is an attempt to increase the explainability of the model, forcing it to predict what we can interpret that is physics properties.

3 Relevant points from guest's lectures

Software Engineering is critical. In my case, SE is not critical but still important since a PhD is mostly individual work. However, projects are usually involved multiple participants, and a level of organization or code management is always appreciated, especially when submitting for publication.

Software Engineering relevance changing with the integration of AI/ML. Not much would be changing in the near future since, as pointed out in the lectures, AI/ML is only a small fraction of the full product, everything else such as data collection, resource management or analysis tools cannot be easily replaced by automated tools.

4 Paper 1: Exploring Hyperparameter Usage and Tuning in Machine Learning Research

4.1 Describe the core idea

The paper addresses the problem of hyperparameter usage and tuning of ML research. Beside algorithm design, selecting the correct set of hyperparameters is crucial to ensure the best possible performance of a ML model. The process involves model's configuration, how the model is trained, data splits, and the randomness in number generation while training. All of this information should always be reported along with the corresponding publication to ensure the reproducibility of the project. As Paper 1 suggests, there exists a lack of documentation of hyperparameter tuning in the recent publications in ML venues, resulting in a need for improved research and reporting practices when developing ML methods. A few key findings have been pointed out in the paper: 1. neural networks provided by PyTorch and TensorFlow are dominating; 2. most hyperparameters in research paper are set to default; 3. hyperparameters are often set as constants; 4. if hyperparameter tuning is adopted, it is rarely documented in detail; 5. computer vision and software engineering have lowest hyperparameter tuning practices.

4.2 Relation to my research

Because I am also developing an ML model for predicting physics properties of human motion, my research also affected by good hyperparameter tuning. When dealing with different datasets, the recording environments and camera setups change, leading to a different sets of hyperparameter for the prediction model. This numerical shift requires intensive tuning process, and usually only sub-optimal results is achieved due to limitation in time and efforts. An systematic procedure for hyperparameter search as suggested in the paper might bring benefit to my research, reducing the spent resource. Also, hyperparameters are also application-specific, which means the same model architecture that is trained to work with different tasks will have different set of hyperparameters. Therefore, each respective application field should spend a few extra efforts to create a baseline hyperparameter tuning procedure, creating a uniform benchmark for research to follow.

4.3 A larger AI-intensive software project

Human motion capture technology can bring benefit to multiple AI-intensive projects. For example, providing high-quality training data for virtual reality and augmented reality AI models. Having better training data can helps the AI model to generate a more realistic and natural animations. Accurate motion capture can be beneficial to rehabilitation and sport analysis, especially with corresponding force estimation, which could help the doctors identifying the medical problem much easier. Another AI-project that can benefit from motion capture is robotics, where accurate motion captures can be used to teach robot how to perform tasks by mimicking the actions.

The key component that ensure the importance of human motion capture technology in these application is accuracy, because only accurate predictions are useful. To ensure the best performance, a suitable hyperparameter search is needed, and it requires a lot of efforts. The method suggested from Paper 1 can be very beneficial in reducing resource on hyperparameter tuning, by proving an overview of the related AI research and help increasing the awareness of hyperparameter tuning documentation.

4.4 How to adapt/change

My own research does not need adaptation since I am fully aware of the important of hyperparameter selection from the beginning. However, there is still a need for a better approach for searching, instead of doing it intensively as currently. Going through all possible hyperparameters of an ML model, especiall deep learning model, is almost impossible right now. So, one way to adapt is to making effort to find the most important ones and set all the rest to default.

5 Paper 2: Towards Understanding Model Quantization for Reliable Deep Neural Network Deployment

5.1 Describe the core idea

The paper addresses the task of quantization for reliable deep learning model deployment. State-of-the-art deep neural networks often contain very large and complex pre-trained models. However, deploying these model in edge devices requires model compression due to the limitation in computational power. Model compression method such as quantization surely reduces performance and accuracy. Due to this problem, the paper conduct a study to characterize the behaviors of quantize models, helping related research to understand more about model compression. Moreover, the data distribution shift when deploying compared to training phase is also addressed, both synthetic and natural shifts. One of the good find of the paper is quantization-aware training can produce stable model than adversarial and Mixup training. Another interesting finding is the *Margin* evaluation metric for uncertainty measurement to distinguish disagreements.

5.2 Relation to my research

My research has not reached this stage of deployment yet, however, this will be very important in the future when we start to create an actual product for human motion capture on hand-held devices. Currently, the developing model is trained on large GPU server with multiple instances in parallel for parameter searching. However, deploying the same architecture and model on edge devices would be very challenging at the current state, and the performance and the processing speed would drop significantly. The suggested quantization-aware training will be a valuable addition to the robustness of the model when deployed on edge devices.

5.3 A larger AI-intensive software project

As mentioned in the discussion of Paper 1, the human motion capture technology fits into multiple different AI-intensive projects such as augmented reality, medical rehabilitation and robotics. Most of the application often stationary machines so the deployment issues would not be a problem. However, in the case of robotics application, especially mobile robots, deployment on edge computers is required. Methods such as quantization or architecture size reduction need to be carefully considered, since they directly affect the performance of the system. To be able to deploy effectively on edge computers, trade-off in performance vs computational speed must be compromised. A reduce in prediction accuracy will drop compared to the developed version, but techniques such as quantization-aware training from Paper 2 could be a solution to retain good performance.

5.4 How to adapt/change

Adapting the quantization-aware training to my current pipeline is not too difficult. However, retaining the same accuracy is the challenging part. Transforming higher-bit level weights to lower-bit level weights can be understood as one type of regularization for the deep learning model. By putting a quite a large constrain on the model, I will need to reduce other regularization parts such as reducing dropout rate, learning rate, and increasing training batch size, model's layers.

The uncertainty metric *Margin* is also a very plausible addition to the model evaluation task. Despite not having a direct consequence to people safety when a fail prediction occurs, taking into account the overly confident situation during deployment is still in demand. We can apply the metric *Margin* real-time as an extra layer of supervision for the model to identify fail cases early.