Software Engineering Assignment

Yushan Zhang
Linköping University

August 31, 2023

1 Introduction to research topic

Computer Vision involves enabling machines to interpret and understand visual information using deep learning methods. It enables a variety of applications in autonomous driving, robotics, medical imaging, augmented reality etc. Typical tasks include recognizing, classification, understanding. To solve these tasks, convolution Neural Networks are usually employed. The sub-task I'm working with is scene flow estimation, which involves estimate 3D motion from a dynamic scene. Two point clouds captured from LiDAR sensor in time step consists of a dynamic scene. The goal of the task is to estimation, for each point from the source point cloud, the motion to the target point cloud. This is done by a matching process. For each point form the source point cloud, we compute a matching score to all the points in the target point cloud. The scores are then normalized to sum to one. The flow vector is then computed as a weighted average of the translation vectors from the source point to the target points. The core to the method is robust features for each point, which is realized by convolution neural networks and transformers.

2 AI/ML support for SE

AI tools for software engineering could help with efficient coding. The techniques include code generation, bug detection, code review, testing etc. All of them can be very useful in computer vision. Code generation can help with generating the skeleton of the neural network model. Bug detection code review, and testing can help with debugging. Making a good use of all the tools can help improve the coding efficiency.

3 SPACE

SPACE represents Satisfaction and Wellbeing, Performance, Activity, Communication and Collaboration, Efficiency and Flow. It is a guide for developing software developers and for improving productivity. This is not only a guide for software development, but also for all the algorithm developers. In computer vision, when developing code, it is also important to ensure the quality and the efficiency. Communication and collaboration with the team members with different expertise will also guarantee the development outcome.

4 Cognitive illusions & biases of our brains

Human brains sometimes work irrationally based on prior information. There has been a long debate about whether Artificial Intelligence should work similarly to human intelligence. Is the cognitive illusions and biases of our brain we would like Artificial Intelligence to learn or not? Personally, I would vote for yes. From the perspective of Computer Vision, if we take the Optical illusion figure as an example, even if square A and square B are of the same brightness, the human brain will recognize square A as darker than square B. This is because the human brain encompasses cognitive abilities and prior information. While AI in computer vision performs well or even better than humans do in specific tasks, e.g. classification, they lack the depth and versatility of human intelligence. As far as I am concerned, there is still a long way to go.

5 ML algorithm & accuracy are NOT enough

Only a small fraction of real-world ML systems consist of ML code. Other aspects including data collection, data verification, machine resource management, analysis tool, and monitoring are also very important. In the field of computer vision, data collection and machine resources are very important and closely related to the ML code. For a deep learning task, especially tasks need to be supervised, it is very important to get accurate annotations during the data collection. Take the scene flow estimation task as an example, if we want to train a model in the supervised mood, we need to collect the ground-truth movement of each point from the first scene to the second scene. This is very difficult to obtain in the real world. What people do is either to approximate the ground-truth or to create synthetic datasets. Computing resources is yet another important aspect to look into in machine learning. For the scene flow estimation task, the estimation may not scale well with more points, requiring more computing resources both during training and inference. Therefore, we need to bear in mind to balance the computing resources and the performances while developing ML algorithms.

6 Human-Computer Interaction

Human-Computer Interaction (HCI) refers to the study and design of the interaction mode between human and digital devices. It is a discipline that involves understanding how people think, behave, and perceive information, and then designing user-friendly and efficient interfaces interfaces that accommodate these factors. The goal of HCI is to produce usable, safe and functional systems that meet the users' needs and preferences. In order to do that, developers gain knowledge of Computer Science, Cognitive Psychology, Social Psychology, Ergonomics/Human Factors, Linguistics, Artificial Intelligence, Philosophy, Sociology & Anthropology, and Engineering & Design. One of the most important concepts in HCI is usability, which is concerned with making the system easy to use. A usable system is defined as: easy to learn, easy to remember how to use, effective to use, efficient to use, safe to use, enjoyable to use. Product designs with bad usability might lead to frustration, wasted time and errors etc. On the other hand, systems with HCI are supposed to be designed as easy, effortless, and enjoyable to use.

Related to Computer Vision, there are some techniques that can be useful for Human-Computer Interaction. For example, 2D/3D facial reconstruction and talking-head generation is a recent technique in computers that can be used to enhance user experience in HCI [3]. With the rapid development of artificial intelligence and machine learning, digital human is fast developed and applied to different fields including personal assistance, intelligent customer service, and online education, etc. This can also be applied to HCI. The goal is to generate interactive objects with natural characteristics. Knowledge of speech recognition, text-to-speech, and video synthesis are required. Among these, computer vision helps with video synthesis, including 2D/3D facial reconstruction, talking-head generation, etc.

In the field of talking-head generation, one of the popular research directions is audio-driven lip synthesis, which involves mapping the lower-dimensional speech signal to the higher-dimensional video signal. The different methods can be divided into 2D-based and 3D-based methods.

In the 2D-based methods, landmarks, semantic maps, or other image-like representations are usually used to solve the problem. In the 3D-based methods, early works employ pre-built 3D models of specific people to solve the problem. However, such methods are usually with high construction cost and difficult to change to a new identity. More recently, 3D model are reconstructed from training videos directly. NeRF (Neural Radiance Fields) is one of the most popular approaches, which simulates implicit representation, and can store 3D coordinate and feature information for high-resolution reconstruction.

7 Architecture and Design

Software architecture is the organization of the overall software system. The importance of software architecture and design emerges as the size and complexity of software systems increases, and that the problem goes beyond the algorithms and data structures. Different architectures and designs including [2]: Pipes and Filters, Data Abstraction and Object-Oriented Organization, Event-based, Implicit Invocation, Layered Systems, Repositories, Table Driven Interpreters, Heterogeneous Architectures, etc. The architecture principles can be used to increase our understanding of software systems. Different architectural solutions to the same problem can provide different benefits.

Software architecture analysis is thus another important orientation to study, which helps with identifying potential risks and verifying the software quality. Two basic classes of evaluation techniques are questioning techniques and measurement techniques. Questioning techniques generate qualitative questions about the architecture, while measurement techniques suggest quantitative measures of the architecture.

When designing computer vision algorithms, we can also bear in mind that the architecture and design are also important besides the algorithm itself. And that some software architecture analysis could be beneficial for the projects. Fran¸cois *et. al.* [1] introduces a new architectural style, Software Architecture for Immersipresence (SAI). It is an architecture model for designing, analyzing, and implementing applications that engage in distributed, asynchronous, parallel processing of diverse streams of data. SAI aims to offer a comprehensive framework for the distributed execution of algorithms and their seamless integration into intricate systems. In the realm of Computer Vision, this aspect of SAI proves particularly beneficial, as it aligns effectively with the common utilization of distributed algorithms and the frequent need for integration into intricate systems. As an illustration, consider the scenario of scene flow estimation from point clouds, wherein individual points can be processed autonomously. In this context, leveraging distributed algorithm execution could lead to a substantial enhancement in efficiency. Hence, the exploration of software architecture and design holds significant importance within the domain of computer vision.

8 Conclusion

With the rapid growth of Machine Learning and Artificial Intelligence algorithms, the importance of ML/AI Engineering increased accordingly. It is not only the algorithms that matter, but more importantly how to implement them in practice. ML/AI Engineering bridges the gap between theoretical concepts in machine learning and AI and their practical implementation in real-world applications. I believe that in the near future, ML/AI Engineering will become an important research area, and bring up many opportunities. As a researcher in the field of computer vision, I would very much like to see ML/AI Engineering facilitate the application of various research outputs into the real world.

References

[1] Alexandre RJ François. Software architecture for computer vision: Beyond pipes and filters. *Inst. for Robotics and Intelligent Systems*, 2003.

[2] David Garlan and Mary Shaw. An introduction to software architecture. In *Advances in software engineering and knowledge engineering*, pages 1–39. World Scientific, 1993.

[3] Rui Zhen, Wenchao Song, Qiang He, Juan Cao, Lei Shi, and Jia Luo. Human-computer interaction system: A survey of talking-head generation. *Electronics*, 12(1):218, 2023.