Control and Analysis for Sequential Information based on Machine Learning

ZHANG, PENG

Control and Analysis for Sequential Information based on Machine Learning

Peng Zhang

A Thesis presented for the degree of
Doctor of Philosophy

Department of Computer Sciences
Durham University
United Kingdom
March 2023

Dedication

I would like to dedicate this thesis to the Future.

Declaration

Parts of this thesis have been taken from published and submitted academic conference/journal papers. All of these papers were primarily written by me, Peng Zhang, during and as a result of my Ph.D. study. Involved papers are listed as follows.

Peng Zhang, Jie Zhang, Yang Long and Bingzhang Hu. An improved reinforcement learning control strategy for batch processes, published in 24th International Conference on Methods and Models in Automation and Robotics (MMAR), IEEE 2019 (Chapter 3).

Peng Zhang, Jie Zhang, Bingzhang Hu and Yang Long. Optimization control of a fed-batch process using an improved reinforcement learning algorithm, published in IEEE Conference on Control Technology and Applications (CCTA), IEEE 2019 (Chapter 3).

Peng Zhang, Yawen Huang, Bingzhang Hu, Shizheng Wang, Haoran Duan, Noura Al Moubayed, Yefeng Zheng, Yang Long. Knowing the Past to Predict the Future: Reinforcement Virtual Learning, submitted to Multimedia Tools and Applications, 2022 (Chapter 4).

Peng Zhang, Bai Yang, Jie Su, Zhenyu Wen, Yang Long. Flow Enhanced Dual Transformer for Video Inpainting, submitted to International Conference on Artificial Neural Networks (ICANN), 2023 (Chapter 5).

Peng Zhang, Bai Yang, Jie Su, Yan Huang, Yang Long. Towards Few-shot Image captioning with Cycle-based Compositional Semantic Enhancement Framework, submitted to International Joint Conference on Neural Networks (IJCNN), 2023 (Chapter 6).

Peng Zhang
March 2023

Acknowledgements

I would like to extend my sincerest thanks to the following, who have all helped in the completion of this thesis.

Firstly, I would like to express my sincerest thanks to my supervisor, Dr. Yang Long, who has provided me with the patient and inspiring knowledge towards the academic field and my life. He is gentle and gives me so much inspiration to explore the mysteries of the machine learning field.

I would also thank my family, especially my parents, sister and brother, Laiyi Zhang, Chunying Cao, Lei Zhang and Kai Cao, who have provided every possible material and spiritual support for my study. The most significant person I would like to thank is my wife, Yaqi Wang. She has spent all her time and energy caring for me and our daughter, Jinyu Zhang, who gives me power in my research. My achievement cannot be apart from their continued love and encouragement. I love you all.

I would also like to appreciate the help and inspiration from my dear colleagues. Jie Zhang, Bingzhang Hu, Yang Bai, and Haoran Duan, as senior members, have provided many helpful supports. Especially, Bingzhang Hu, my friend, offered many helpful supports in both my research and living. Also, I have collaborated with Shizheng Wang, Yan Huang, Likun Qin, Jie Su, Junyan Wang and Fan Wan with loads of joy. I will extend my sincere thanks to Li Lei, Zhengyu Zhao, Yichen Zhu and Chen Cui, all of whom could not be fully listed here thanks to all of you.

Finally, I thank Durham University for supporting the professional research platform and beautiful environment, making me have happiness research life.

Abstract

Sequential information is crucial for real-world applications that are related to time, which is same with time-series being described by sequence data followed by temporal order and regular intervals. In this thesis, we consider four major tasks of sequential information that include sequential trend prediction, control strategy optimisation, visual-temporal interpolation and visual-semantic sequential alignment. We develop machine learning theories and provide state-of-the-art models for various real-world applications that involve sequential processes, including the industrial batch process, sequential video inpainting, and sequential visual-semantic image captioning. The ultimate goal is about designing a hybrid framework that can unify diverse sequential information analysis and control systems

For industrial process, control algorithms rely on simulations to find the optimal control strategy. However, few machine learning techniques can control the process using raw data, although some works use ML to predict trends. Most control methods rely on amounts of previous experiences, and cannot execute future information to optimize the control strategy. To improve the effectiveness of the industrial process, we propose improved reinforcement learning approaches that can modify the control strategy. We also propose a hybrid reinforcement virtual learning approach to optimise the long-term control strategy. This approach creates a virtual space that interacts with reinforcement learning to predict a virtual strategy without conducting any real experiments, thereby improving and optimising control efficiency.

For sequential visual information analysis, we propose a dual-fusion transformer model to tackle the sequential visual-temporal encoding in video inpainting tasks. Our framework includes a flow-guided transformer with dual attention fusion, and we observe that the sequential information is effectively processed, resulting in promising inpainting videos. Finally, we propose a cycle-based captioning model for the analysis of sequential visual-semantic information. This model augments data from two views to optimise caption generation from an image, overcoming new few-shot and zero-shot settings. The proposed model can generate more accurate and infor-

mative captions by leveraging sequential visual-semantic information.

Overall, the thesis contributes to analysing and manipulating sequential information in multi-modal real-world applications. Our flexible framework design provides a unified theoretical foundation to deploy sequential information systems in distinctive application domains. Considering the diversity of challenges addressed in this thesis, we believe our technique paves the pathway towards versatile AI in the new era.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Sequential information is significant in the real-world applications related to time. It is same with time-series, which is described by sequence data followed by temporal order and regular intervals. Meanwhile, the various platforms create a large amount of sequential information contributing approximately 90% of the total data of the world [1]. Sequential information is a multi-dimensional description. As shown in Figure 1.1, industrial process [2], computer vision [3], robotic control [4] and language process [5] etc. analyse and provide sequential information with different perspectives. The illustration of sequential information is diverse for various fields. For example, some works summarise a function to simulate a chemical process to simplify the control process through observation of the produced sequential data [2]. In another example, the camera can recognise a specific object by analysing each consecutive frame of a video sequence. Exploiting sequential information in the language process benefits the development of automatic speech recognition. In addition, it is promising that the control strategy provides a human-like interaction for the robotic domain, attributing to the collection and analysis of sequential information. Generally, sequential information includes temporal dependencies leading to identical data points having different behaviours at various times. Existing efforts

control and interpret the time-dependency through amounts of hand-crafted experiments, which are expensive to obtain and highly rely on costly professional domain knowledge of different areas.

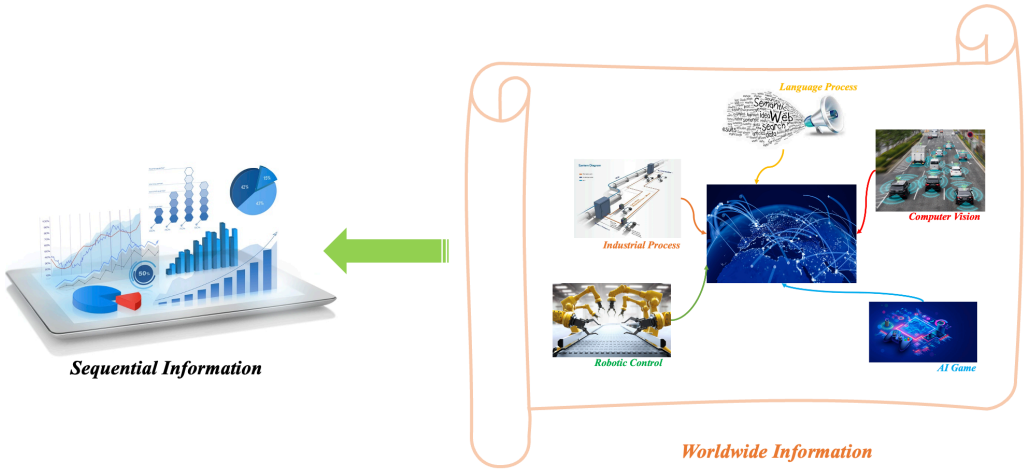


Figure 1.1: The sequential information from the world.

With the development of Artificial Intelligent (AI) technology, sequential information could contribute to more value through research works. As the most popular theory, artificial intelligent technology has been widely applied in computer vision since its origin in 1943 [6]. Contemporary AI research manipulates artificial neural networks to model and predict sequential information. Fewer works focus on the control of sequential information. Considering the power and potential of AI technology and the importance of sequential data worldwide, it is a new opportunity and challenge to combine AI and traditional sequential information analysis and control the a unified theoretical framework.

1.1 Sequential Batch Process

Our work starts from a conventional and classic field for the industrial process which has witnessed multiple revolutions due to new technologies. The batch process is the most vital part of the chemical process, which is used to generate small quantities of high-value productions such as pharmaceuticals, fine chemicals and polymers [7]. For example, fermentation is the most key part in the production process of Penicillin. As reactant, glucose (Raw Material) and penicillium (Reaction

Material) can prodcue Penicillin (Product), meanwhile, fermentation effluent (By-product) is also produce during this process. Therefore, the key goal is to maximise the Penicillin while minimising fermentation effluent, which is shown in Figure 1.2.

Recently, more chemical industries have increased the demand for the quality of different products. Hence batch manufacturing is acquiring increased demand. Generally, the challenge of batch process manufacturing is that the production still has a very narrow range of quality requirements under the apparent distinction of feedstock. The sequential control strategy with respect to the process effectiveness is yet agnostic.

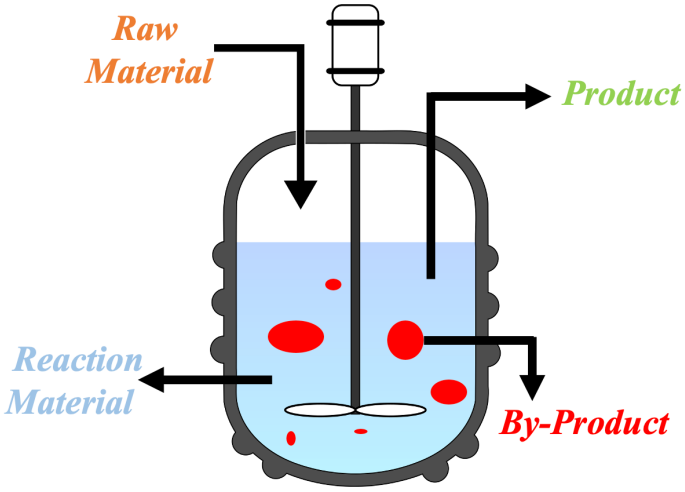


Figure 1.2: The description of the batch process. The key control parameter is the rate of raw material. The objective is to maximise the product while minimising the by-product.

Due to the imperative demand, the design of the control model is vital for the efficiency of the whole industrial process. Many efforts apply traditional advanced control methods for the batch process. But these methods are slow compared with continuous counterparts [7]. The primary reason is that the operation of the batch process has a nonstationary character which involves the strong non-linearity of the dynamics chemical process. Although non-linearity control methods have been created, the computational complexity and costly prior domain knowledge for the implementation often obstacles to real industry execution. Besides, a model of the nonlinear process is challenging to improve, while the increased complexity of implementation in the industry area.

Control strategy mainly focuses on making dynamic decisions in sequential observations. In this thesis, we investigate the simple impact of feed rate in the batch process. While most conventional control methods of batch process adjust the control strategy through repetitive experiments based on process simulation, the exploration is limited to human experience and time-costly. Dynamic optimisation of control parameters using fewer empirical data is a critical and imperious requirement. The hybrid of multiple control technology has significant potential improvement in batch process control. The feed rate affects the concentration of each element and volume dynamics in the container [8]. The non-optimal feed rate damages the balance of the reaction. Additionally, the feed addition changes the total volume, which directly responds to the concentrations of all products to impact the balance of the reaction. Overall, the feed rate affects the system in several ways. There are different control objectives defined by different motivations. In many industrial processes, maximising desired product concentration is the critical control goal. An optimal feed rate can saturate the pathway for product formation to generate the high desired product concentration. The process is difficult to achieve maximum productivity if too little is fed. On the other hand, overfeeding leads to overflow metabolism that produces the undesired by-product [8], shown in Figure 1.2. Because of the trade-off, the resource cost is a considerable challenge in the batch process. The control methods not only balance the concentration between the desired product and the undesired by-product but also minimise the wastage of the raw material.

1.2 Sequential Information in Computer Vision

Vision, as a natural and important ability of human beings, provides abundant information such as colour, shapes and relationships of objects etc. Human develops exciting knowledge by integrating vision information. In computer vision, high-level semantic information is processed through extraction, analysis and understanding from digital images and videos. Wide applied computer vision techniques in the industry include games, healthcare, etc. Extracting visual information has crucial improvements in computer vision tasks before the deep learning technique experi-

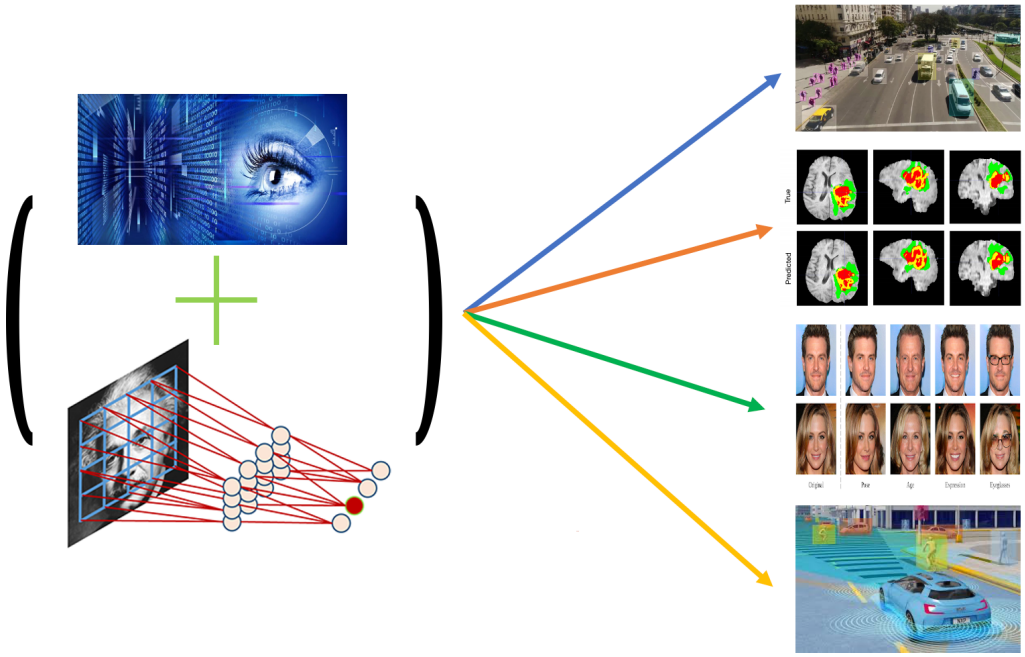


Figure 1.3: The development of computer vision based on deep learning.

ences tremendous progress.

Over the last few years, deep learning has been demonstrated to improve the benchmark of various fields. Computer vision is one of the most prominent applications [9] due to the complex data structure by nature. According to Figure 1.3, many computer vision problems have been boosted by deep learning, such as object detection, face recognition, medical recognition, action recognition and image generation etc. With all of these improvements, convolutional neural networks (CNNs) provide important technical foundation. CNNs was first proposed and inspired by the visual structure in 1962 [10]. Neocognitron proposed the first computational model to acquire the transformation of images based on local connections of neurons and hierarchical organisation [11]. This model indicates that patches of the previous layer at different locations execute neurons with the same parameters, which obtain a form of translational invariance [9]. Following this inspiration, convolutional neural networks applying the error gradient designed by Yann LeCun. Improved performance in a wide range of computer vision tasks has been observed [12] [13] [14].

Despite the unprecedented revolution of discriminative deep models, image generation is considered a challenging research topic in computer vision. Images from social networks describe relationships between each object, including the main char-

acter, scenes, location and time. Minor modifications of the visual contents can change the semantics of the images. Digital images are usually modified for various reasons, such as scratch and text removal, object removal and random mask. Hence, image inpainting, as a typical control task of image generation, has caused increasing attention in recent years. The goal of image inpainting aims to control the visual content generation to fill the missing region of the image. The challenge is to infer the semantic information of missing regions from surrounding regions in an image.

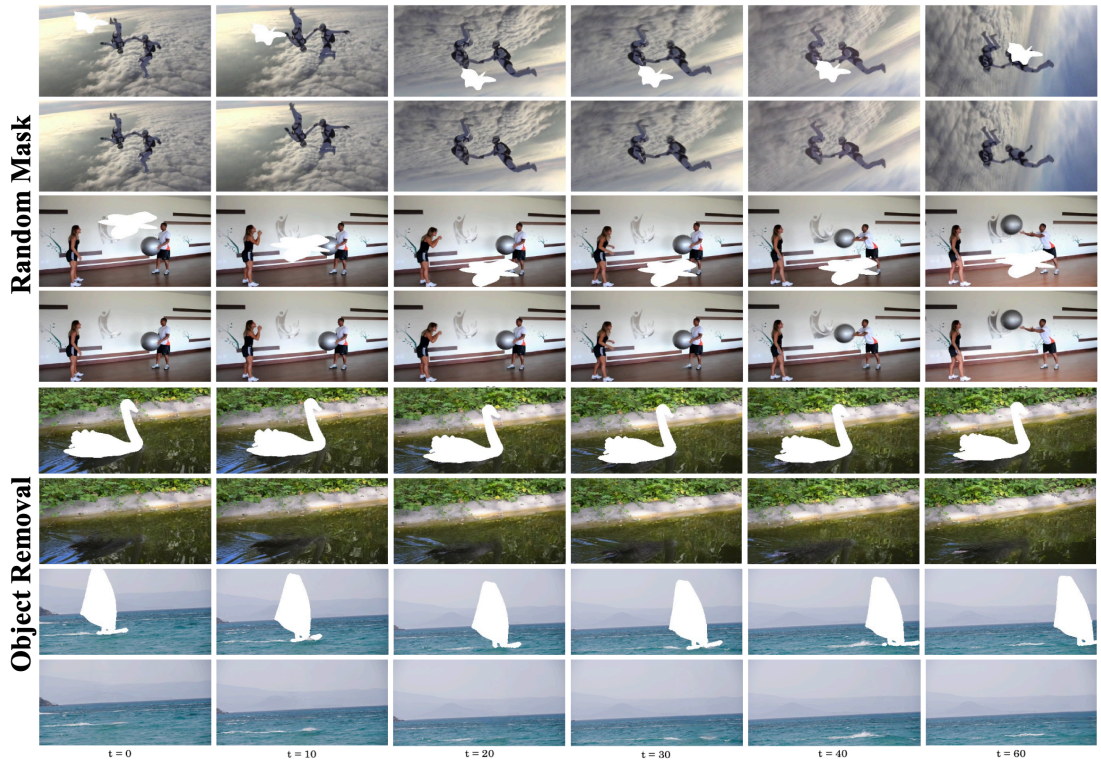


Figure 1.4: The examples of video inpainting.

Compared to image inpainting, video inpainting has more challenges. Additional sequential information brings complicated motions and the output requires high temporal consistency [15]. Figure-1.4 shows examples of video inpainting with random mask and object removal. The first two rows show examples with random masks. The last two rows show object removal, i.e. the entire foreground object is removed. Besides completing the missing regions of each frame, the methods must ensure to guarantee temporally consistent [16]. Although some efforts directly apply deep learning-based image inpainting to address video inpainting tasks, these

approaches suffer from issues. For instance, using an image inpainting approach on each frame of the video leads to jitters and temporal artefacts. In the meantime, it is hard to obtain sufficient sequential coherence when the input is a long video sequence. Finally, there is high computational consumption when controlling and generating reasonable content that completes different missing regions at each frame. Because of these difficulties, efficiently controlling and analysing sequential video information is still challenging.

1.3 Sequential Visual-semantic Models

Different types of data such as images, texts, audio and videos can describe and observe things from different perspectives. For instance, the description of a specific event from the internet usually contains texts, videos and images. Generally, different types of data with heterogeneous properties cannot be processed using a unified multi-modal data structure. Due to heterogeneous feature spaces, multi-modal data has attracted substantial attention recently. While each multi-modal data represents a distinct property, they are complementary to each other [15]. Recently, deep learning approaches have obtained the nonlinear distribution of high-dimensional single view. Leveraging the information from multi-modal may boost the overall performance better than each of the single modalities. According to [17], this hypothesis is theoretically proved that multiple views provide more information than the sum of every single view and to improve task performance.

Overview of cross-modal Cross-modal learning aims to take a main type of data to describe the other modalities. The biggest challenge due to the heterogeneous property of multi-modal data is how to effectively capture the semantic and correlation between data modalities, i.e. the heterogeneity gap [18]. Figure 1.5 illustrates the challenge of cross-modal representation learning: image, video, audio and text indicate different modalities. Using specific nonlinear features through the deep learning model, all the representation features are projected in a common subspace to capture the joint distribution.

The visual-semantic cross-modal tasks include image-text retrial, Visual Ques-

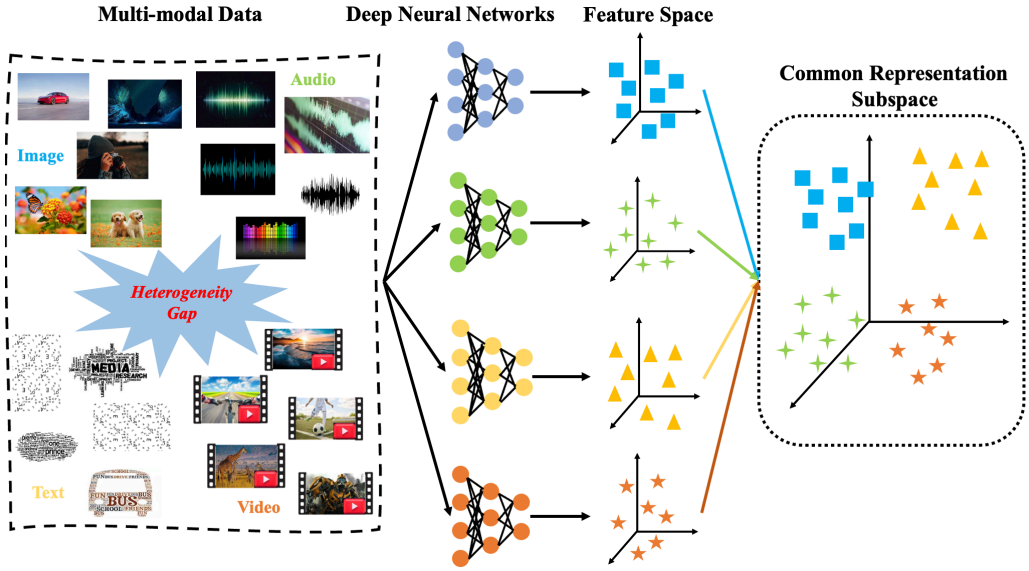


Figure 1.5: The challenge of multi-modal data.

tion Answering(VQA) and image captioning. Automatic caption generation is essential in education, digital libraries and web searching etc. Image captioning focus on generating a description through an image. Specifically, image captioning requires extracting the objects, attributes, scene type, location and related information in an image, generating well-formed sentences to match the syntactic and semantic understanding of the language [19]. Image captioning has more challenges than other visual-semantic tasks for two main reasons. Firstly, the model must not only extract the representation of each object in the image but also can obtain the complex relationship via inference. Secondly, generating descriptions relies on sequential information, i.e. the current generated content is highly dependent on previous outputs. Although existing deep learning-based models achieve state-of-the-art performance in image captioning, models still struggle when the data quality is not reliable in the few-shot and zero-shot settings.

1.4 Contributions and Thesis Outline

The contributions of this thesis overcome the above challenges and propose unified frameworks for diverse applications and tasks while maintaining state-of-the-art performances on each of the tasks. The rest of the thesis is organised as below:

Chapter 2: Background An comprehensive literature review will be provided for hybrid control and analysis for sequential information and the applications in process control, video inpainting and image captioning.

Chapter 3: Sequential Control Process with Reinforcement Learning In this chapter, we investigate the control approach based on deep learning to the batch process of the chemical process. Firstly, we apply reinforcement learning to design the control model aiming at two cases of the batch process. Then, we proposed two improved reinforcement learning algorithms to control the different batch processes, respectively. Finally, the control results of reinforcement learning are compared with traditional methods to prove the advance of our methods in the process control field.

Chapter 4: Hybrid Reinforcement Virtual Learning for Sequential Control Process Based on the investigation from chapter 3, we focus on further optimisation of the control strategy for the batch process. This chapter proposes a hybrid reinforcement virtual learning framework for high-efficiency learning. Hybrid reinforcement virtual learning is a flexible framework, including virtual space and reinforcement learning control elements. The first contribution is that the virtual space executes historical industrial process data to build a virtual environment. In this space, the reinforcement learning part can interact with the virtual environment to obtain a virtual control strategy without interacting with any real environment. Then, when the reinforcement learning part interacts with the real environment, the virtual control strategy guides the reinforcement learning to learn the real control strategy when interacting with the real environment. The second contribution is that the virtual space predicts several future control results after interacting with the virtual environment to optimise the virtual strategy. The reinforcement learning part can efficiently obtain a real strategy based on the optimal virtual strategy. The final contribution is that the control results of the proposed framework prove that hybrid reinforcement virtual learning achieves state-of-the-art performance.

Chapter 5: Sequential Visual Information in Video inpainting In video inpainting, as a typical visual sequential information task, we propose a dual-fusion transformer to optimise the control of inpainting. In this chapter, the first con-

tribution is that applying the optical flow provides extra sequential knowledge to overcome temporal consistency issues in video inpainting. Second, we propose an attention-wised transformer model to fuse two views reasonably, leading that features of two different views interact to obtain sufficient spatial-temporal output. Benefits from these contributions, the proposed model can control the visual contents to complete the missing regions by time series. Finally, the proposed dual fusion transformer model achieves state-of-the-art performance.

Chapter 6: Sequential Visual-semantic Information Analysis with Cycle-based Framework We focus on image captioning to investigate the sequential visual-semantic information analysis in this chapter. Due to the heterogeneity between semantic information and visual information, it is challenging to connect and interact with each other. Furthermore, the few-shot and zero-shot multi-modal tasks are more difficult based on the above challenges. In order to handle these issues, we proposed a cycle-based captioning framework to optimize image captioning. Specifically, the proposed cycle-based framework and switcher module can augment data twice, which means that more data train the model. Meantime, aiming to image captioning, we investigated the popular dataset to define the new few-shot and zero-shot settings. Finally, the experiments indicate that our framework can optimize the visual-semantic information while achieving state-of-the-art performance on the new few-shot and zero-shot settings.

Chapter 7: Conclusion and Future Work Based on above investigations of sequential information, we summarise the contributions of different sequential tasks in this chapter. We further explore future works aiming at sequential process control, sequential computer vision and sequential visual-semantic.

CHAPTER 2

Background

Sequential information is highly related to order and time. There are different ways to model sequential processes and the tasks are diverse. Since it is impossible to cover wide applications of sequential modelling, this thesis focuses on providing representative examples in typical domains. Our hybrid methodology combines both traditional approaches and deep methods. This chapter mainly reviews related research work on hybrid control and analysis for sequential information. Firstly, we investigate traditional control methods and recent deep approaches for batch process control. Then, we focus on reviewing the deep-learning-based efforts for video inpainting. Finally, we review image captioning with more focus on the visual-semantic sequential information.

2.1 Process control

Overview Process control is one of the simplest sequential control problems which has wide industrial applications. In this thesis, we focus on the challenging nonlinear time-varying process control. Both traditional and deep learning-based approaches are reviewed.

2.1.1 Traditional control strategy

Adaptive control methods As a typical non-linear control algorithm, adaptive control strategies have been extensively applied in batch process [20], which adapts the controller parameters over the operation. The common methods of adaptive control are that certain parameters can change in the whole process to better solve the non-linear dynamics and uncertainties in the system. Hisbullah et al. [21] applied the gain scheduling to simulate the yeast system in the batch process, where the controller gain is adapted with prior knowledge to address the changing dynamics. Furthermore, Duan et al. [22] and Jenzsch et al. [23] proposed the hybrid adaptive gain scheduling method to improve the adaptation on non-predictable system dynamics, which applied artificial neural network (ANN) to predict the state variables that unable to be measured in-line. In a different method of adaptive control, the model reference adaptive control (MRAC) were used by Oliveira et al. [24] [25] and Landau et al. [26] in the batch process. In this line, MRAC defines a control action aiming to minimize the error between the provided output from the control model and the actual process output.

Fuzzy Control The inspiration for fuzzy control comes from the principles of fuzzy logic, which mainly addresses uncertain systems without complex models. Because of the non-linear of the batch process, uncertainty often arises in the produce process [27]. Fuzzy control illustrates the engineer's experience with the process controlling the system under evaluation from the current state of the process. Based on this direction, Verbruggen et al. [28] applied the fuzzy control strategy without model identification to solve the complex non-linear process, which is more intuitive to the user based on the combination between user knowledge and past data [29] [30]. Although some research works applied fuzzy control in the batch process, the strong dependence on user knowledge limits the implantation in the batch process.

2.1.2 Artificial Neural Networks -based control

Recently, artificial neural networks (ANN) experienced an improvement due to the exploration of big data. As a typical data-driven technology, ANN, based on his-

torical data can establish a complex non-linear system [31]. To obtain the desired control strategy by ANN, Holland [32] proposed a genetic algorithm that optimised the determination of an optimal value aiming at a certain variable. Due to the benefits of ANN, several efforts have widely applied ANN in the batch process. For instance, Ferreia et al. [33] combined ANN as a predictor for a variable of the system with a feedback control approach to control the batch process. Alternatively, Chen et al. [34] and Peng et al. [35] directly applied ANN in the optimisation period. Furthermore, Cavagnari et al. [36], Chen et al. [37] applied ANN to directly estimate the relationship between system output and optimal input by data acquired from the simulation of model predictive control (MPC).

Different to previous work, our work focuses on deep reinforcement learning for the long sequence control processes. The multiple-step learning for long-term planning and prediction is particularly useful in the control process. Another breakthrough is the design of the hybrid reinforcement virtual learning framework. A long-term trend predictor is trained to predict the control consequences and guide the reinforcement learning model to make better decisions. The framework is very flexible for simulator-free scenarios, i.e. there is no empirical model to guide the training of reinforcement learning. Instead, the trained neural predictor provides a virtual environment. And we believe this is the first-ever neural digital twins system for process control.

2.1.3 Reinforcement Learning

In recent years, machine learning has experienced rapid development; for example, neural networks and deep learning have been applied in many areas [38] [39]. As one conventional and classic machine learning algorithm, reinforcement learning is an advanced control algorithm and its theory is inspired by behaviourist psychology. It follows the Markov decision process (MDP) which includes state S, action A, transmission probability P, and reward function R [39]. Reinforcement learning has been widely applied in multi-agent systems, control theory, information theory, operation research, game theory, simulation-based optimization, genetic algorithm and swarm intelligence [39].

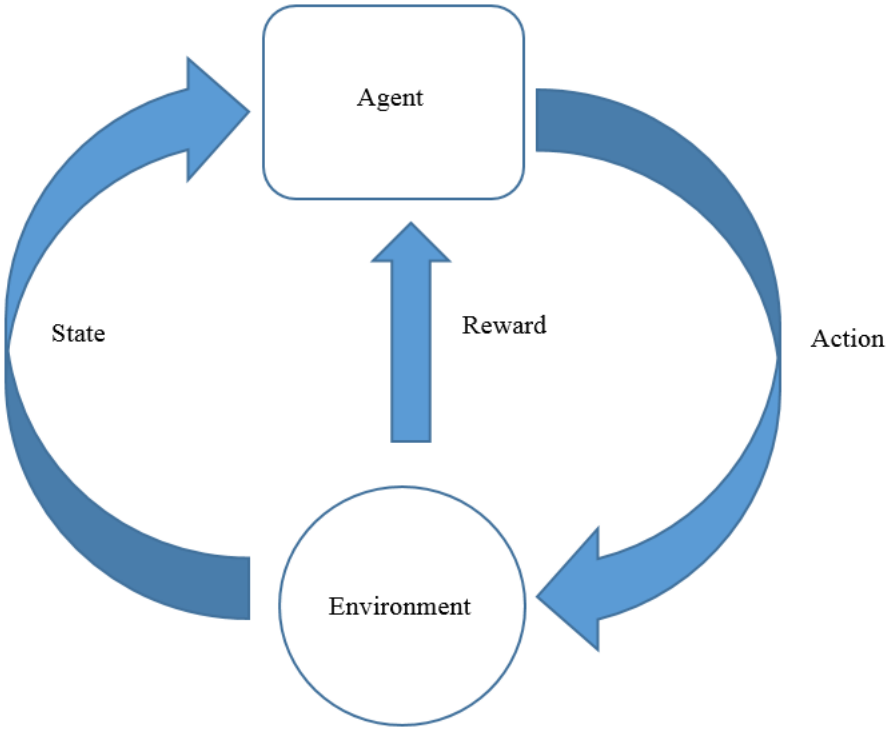


Figure 2.1: Structure of reinforcement learning agent can be broken down into four main components: the agent, the environment, the state, and the reward signal.

For reinforcement learning, an action will be explored and chosen to reach a new state after interaction between the decision-making agent (intelligent controller) and goal dynamic system (environment) and then the reward is calculated [40]. According to the reward function, plenty of special preferences information based on penalties and rewards will be collected in reinforcement learning [39]. Figure 2.1 shows the structure of reinforcement learning. The agent is the entity that interacts with the environment. It takes actions based on its current state and receives feedback from the environment in the form of a reward signal. The agent's goal is to learn a policy that maps states to actions that maximise the cumulative reward over time. The environment is the world in which the agent operates. It provides the agent with feedback in the form of a reward signal based on the actions taken by the agent. The environment can be a physical system, a simulated environment, or a game. The state refers to the current situation or context in which the agent is operating. The state is represented by a set of variables that describe the current state of the environment. The agent uses the state information to decide what action to take.

The reward signal is the feedback that the agent receives from the environment after taking an action. The reward signal is a scalar value that represents how good or bad the agent's action was in achieving its goal. The agent uses the reward signal to update its policy so that it can take better actions in the future.

Traditional Q-learning algorithm There are many algorithms for reinforcement learning. Among them, Q-learning is the most well-known algorithm. As a classic reinforcement learning algorithm, the Q-learning is a model-free algorithm for implementing dynamic programming (DP), which means that the agent of Q-learning can optimally act in Markovian domains [39].

For Q-learning, the main task is to estimate the cumulative future reward, which can be used to select the action in each visited state [39]. To weigh more heavily on the near-term rewards, the discount factor γ will be applied and the reward function (R_t) is maximized as the main objective of agent [39], as shown in:

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k} \tag{2.1}$$

where r_{t+k} is the immediate reward signal. The action-value function $Q(s_t, a_t)$ describes the expected value when selecting an action at state s_t:

$$Q(s_t, a_t) = E\left\{R_t | s_t = s, a_t = a\right\} \tag{2.2}$$

This is the most important key point in this algorithm. This Q-function can describe the benefit when the agent chooses an action a_t at state s_t. As well as, the agent will choose an action a_t after observing the current state s_t; then, the next new state s_{t+1} will be achieved. At that moment, the new immediate reward signal r_{t+1} can be collected. The function, hence, can be obtained by the sum of rewards with the estimated value function [39]:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_t \left[r_{t+1} + \gamma \max_{b \in A_{s_{t+1}}} Q(s_{t+1}, b) - Q(s_t, a_t) \right] \tag{2.3}$$

Here, the next reward is r_{t+1}, b represents the possible available actions in the next new state and $\max Q(s_{t+1}, b)$ is the maximum Q-value in state s_{t+1}. The

discount factor is described by $\gamma(0 < \gamma < 1)$ deciding the effect of short-term and long-term rewards. Besides, the learning speed of Q-learning is controlled and optimized by learning rate $\alpha(0 < \alpha < 1)$ [39]. Additionally, the $\varepsilon - greedy$ policy can be used to decide the exploitation and exploration. Exploitation means that if the agent knows the optimal value function $(\max Q(s_{t+1}, b))$ then greedy action can be selected by the policy. On the contrary, exploration means that if the agent does not know the optimal value function, then it will explore and choose the action of the optimal function [39]. And different updated Q-values can be acquired and these Q-values can form a Q-table. The agent can follow this Q-table to choose the next action.

With the improvement of reinforcement learning, the multi-step action Q-learning algorithm (MSA) is proposed [41]. Based on traditional one-step methods, the rewards of multi-step Q-learning (MSA) are considered from multiple steps [42]. MSA improved the performance of classic Q-learning by combining experience replay [43–45]. Besides, MSA algorithm bootstrapped over long time intervals to improve the traditional one-step TD algorithm [46]. The main difference between MSA and traditional Q-learning is the selection of action. For the traditional Q-learning, the agent selects a new action immediately in the current state in each step time [39]. However, for the MSA algorithm, the semi-Markov option is utilised, which means that the agent cannot explore new action in every state. On the contrary, the agent will continuously apply the same previous action in m time steps to acquire the next new state [39]. For instance, if this degree $m = 3$, the agent will select the same action for 3-time steps in each period as shown below:

$$s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_1} s_3 \xrightarrow{a_2} s_4 \xrightarrow{a_2} s_5 \xrightarrow{a_2} s_6 \ldots$$

$$\ldots s_{i-1} \xrightarrow{a_i} s_i$$

where s_i indicates the ith goal state. Through this improvement, an action can be repeatedly executed for consecutive number of time steps. The Q-value function of

MSA is shown as [39]:

$$Q^m(s_i, a_i^m) \leftarrow Q^m(s_i, a_i^m) + \alpha_t \left[R + \gamma \max Q(s_i, a_i) - Q^m(s_i, a_i^m)\right] \tag{2.4}$$

2.2 Sequential information in Video inpainting

Overview Vision is one of the most modalities in AI perceptive systems. Sequential information is critical for visual analysis. Video inpainting shares the challenges of the image inpainting task. In this thesis, we focus on processing and analysing sequential information in video inpainting. Several works have focused on completing missing regions with temporal and spatial coherent information in the video sequence [47] [48] [49] [50] [51]. In this thesis, we mainly review deep learning frameworks with a path-based strategy.

2.2.1 Path-based Strategy

In the early work, video inpainting was modelled as a patch-based optimisation task [52] [53] [54] [55]. Specifically, these approaches sampled similar spatial-temporal patches from the available areas to generate the missing regions [49] [55]. Meanwhile, other methods applied foreground and background segments to improve the performance [54]. Despite promoting the video inpainting performance, patch-based methods assume a homogeneous motion in the missing regions, while the complex motion is the general situation. Additionally, the optimisation of patch-based methods often suffers from high computational, which is challenging to apply in real-world applications [51].

2.2.2 Deep learning Strategy

Due to the challenges of patch-based methods, many works executed deep learning model in the video inpainting, which boots the performance of this task. This direction can be divided into three categories: 1. 3D convolution-based models; 2. flow-based models; 3. attention-based and Transformer models.

3D Convolution-based Models For 3D convolution-based models, some works exploited 3D convolution and attention strategies to solve the temporal inconsistent issue. Based on this line, the mechanism of combination between 2D and 3D convolution was proposed by Wang et al. [50] to inpaint the missing regions in a video sequence. Kim et al. [56] applied a recurrent neural network (RNN) to integrate the sequential representations by traversing all video frames. Lee et al. [57] designed a copy-and-paste network to learn the corresponding features from known video frames; then pasting them to complete the missing regions in the target frame. Change et al. [58] proposed a Gated Temporal Shift Module and modified gated convolution to the 3D version for improving the free-form video inpainting [59]. Zhang et al. [60] provided a training strategy, namely one-size-fits-all model, to use in different video sequences through applying the internal learning strategy. Furthermore, Hu et al. [61] created a region proposal approach to select the best-completed contents from all participants.

Flow-based models Optical flow can provide motion information that assists many visual sequential tasks, such as video segmentation, video object detection, video understanding, video super-resolution, etc. Hence, several research efforts applied optical flow as extra prior information in the video inpainting to further overcome the inconsistent temporal issue [56] [62]. However, the invalid regions from video frames are the occlusion factors, meaning that directly computing the optical flows of these regions themselves can limit the task performance. Recently, the flow-based models usually completed the optical flows of video sequences first, then propagating indexed pixels based on trajectories of the predicted optical flows [63] [51]. Specifically, most video inpainting models exploit optical flow directly aggregated flow-based information with traditional spatial information [64] [65] [66] [67].

Attention-based and Transformer Models Although these approaches integrate the spatial information from neighbour video frames, capturing spatial knowledge from long-range frames is extremely difficult. To effectively increase the model's ability for long-range correspondences, some works adopted attention mechanisms in the image inpainting and video inpainting [68] [69] [70]. Based on the attention-based

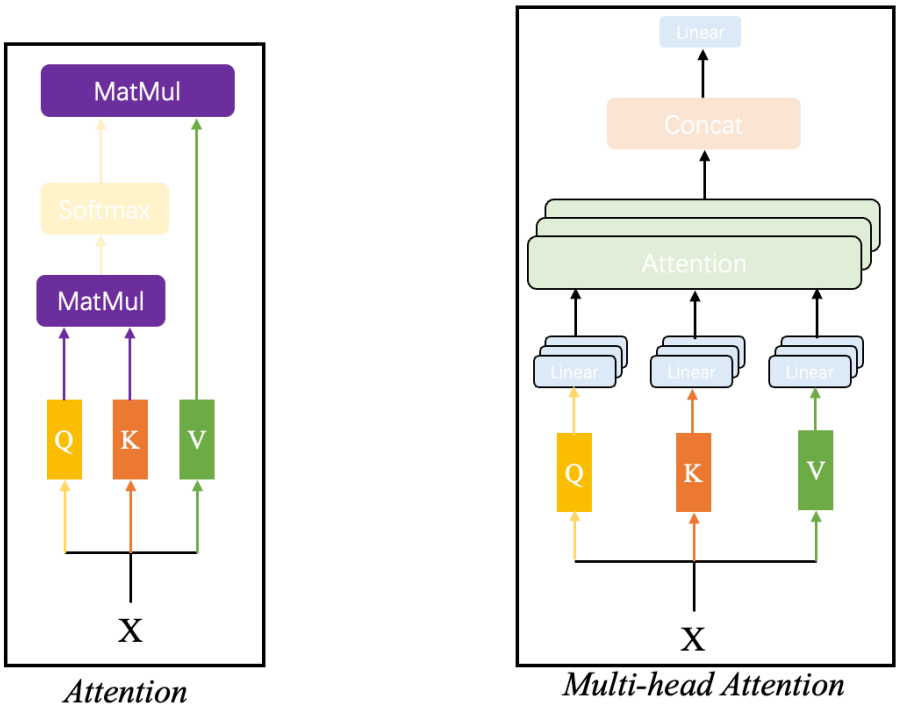


Figure 2.2: Structure of attention

strategy, Li et al. [71] executed temporal consistent information to propagate to the target video frame with the dynamic long-context aggregation attention mechanism. Furthermore, more and more works have applied the Transformer-based strategy in video inpainting. Significantly, Vision Transformer and its variants [72] [69] achieved impressive improvements in the video inpainting task. Due to the quadratic complexity of self-attention, several efforts proposed the window-based attention modules to increase the computational efficiency [73] [74].

Single Attention The attention is the key part in Transformer model, which can be represented by mapping a query and a set of key-value pairs to an output. Packed a set of queries simultaneously together into a matrix Q compute the attention function [75]. Similar, the matrices K and V are constructed by packed the keys and values together [75]. The matrix of outputs as:

$$Attention(Q, K, V) = Softmax(QK^T/\sqrt{d_k})V \tag{2.5}$$

Here, $\sqrt{d_k}$ is the scaling factor. As the most common attention function, dot-product attention is much faster and high-efficiency in experiment, it can be applied by highly

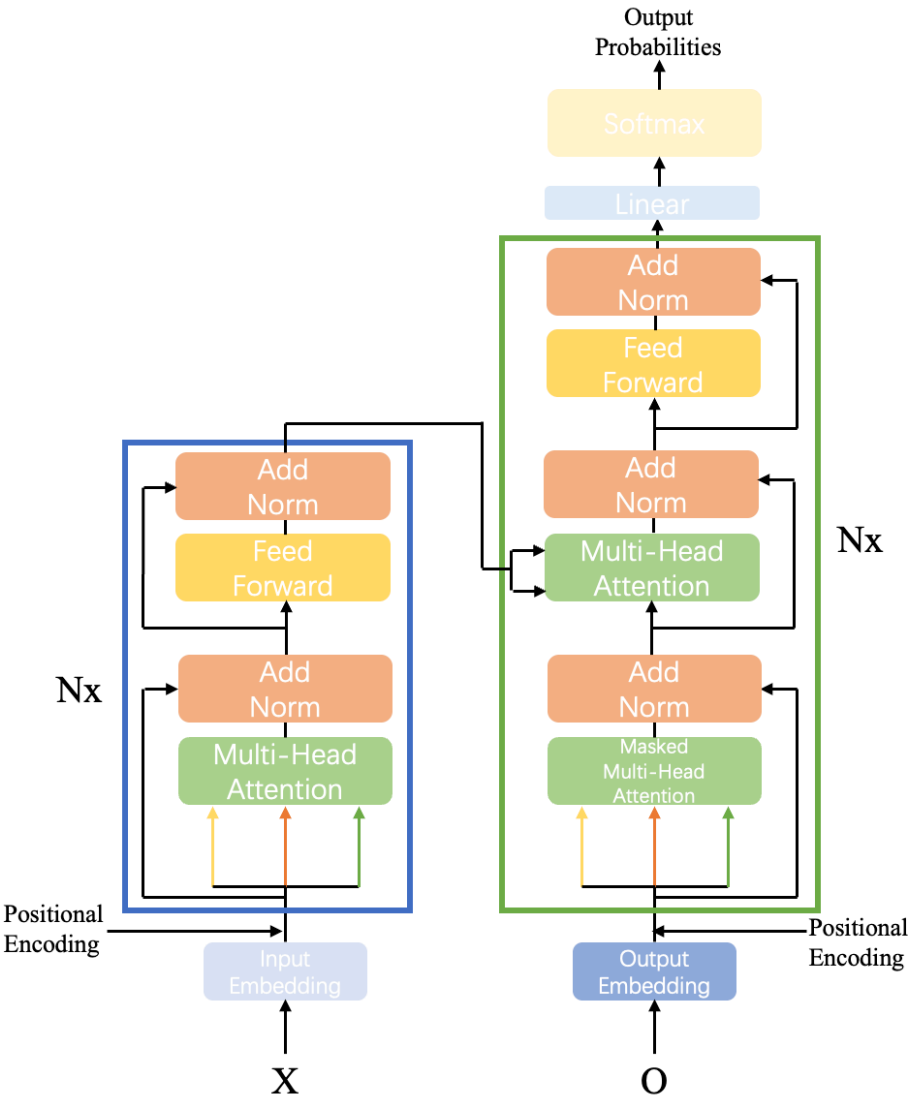


Figure 2.3: Structure of Transformer

optimized matrix multiplication [75].

Multi-Head Attention Based on single-attention, it is benefit to linearly project the queries, keys and values with h times, which can learn projections to d_q, d_k and d_v dimensions [75].

$$MultiHead(Q, K, V) = Concat(head_1, ..., head_h)W^O \tag{2.6}$$

Here, W^O is the linearly projection. The projected versions of queries, keys and values can be performed the attention function in parallel with d_v dimensional output vales, which are concatenated to obtain the final values [75].

Transformer Model The Transformer is constructed by the encoder and decoder with stacked self-attention and fully connected layers [75]. The encoder consists of a stack of N=6 layers. The self-attention and position wise fully connected feed-forward network construct each sub-layers. A residual connection was applied in each of two layers [75]. The decoder is similar with encoder based on a stack of N=6 layers. However, the decoder has a third sub-layer, which is multi-head attention over the output of the encoder stack [75]. In addition, the masked multi-head attention in the decoder ensures that the position of predictions only rely on the know outputs [75].

2.3 Sequential Visual-Semantic Information

Overview Language models and semantic information is also highly related to the sequential information of the data representation. Existing image captioning methods include main three categories: 1. retrieval image captioning; 2. template-based image captioning; 3. novel image caption generation. In this thesis, the main investigation direction is deep-learning-based approaches. Most deep-learning-based methods are included in the category of novel image caption generation [19]. Hence, we mainly investigate novel image caption generation based on deep learning in this thesis. As the typical multi-modal task, image captioning includes visual and semantic perspectives. Specifically, the representations of both two perspectives are the most crucial elements. Therefore, we mainly review the developments of visual encoding and language models in the image captioning task.

2.3.1 Visual Representation

Obtaining a representation of visual content is the first challenge in image captioning. The current visual encoding methods can be summarised in four main directions: 1. global convolutional neural network (CNN) features; 2. attention over regions of CNN features that encoded visual information applying visual regions; 3. graph-based approaches combining relationships between object regions of visual content; 4. self-attention algorithms based on transformer-based model.

Global CNN Features With the improvements of CNNs, several tasks containing visual information have improved in recent years. For image captioning, the model usually executed the activation of one of the last layers of a CNN to acquire high-level representations applying to the language model [76], which was explored in "Show and Tell" paper [77]. In this work, the global CNN features, as the input, were fed to the initial hidden state of the language model. Meantime, Karpathy et al. proposed a similar method using global CNN features to the language model [78]. Furthermore, Donahue et al. [79] and Mao et al. [80] applied the extracted global visual features at all time steps of the language model. Then, kinds of image captioning works widely employed global CNN features [81] [82] [83]. Notably, Rennie et al. [84] proposed the FC model in image captioning, which indicates that implementing ResNet-101 [85] extracted images to obtain their original dimensions. Besides, some methods represented the most common object words of training captions by probability distribution through integrating high-level attributes [86] [87]. Although the image captioning model can simply use global CNN features, the methods with this direction excessively compress visual information. Hence, the approaches with global CNN features are challenging to generate fine-grained and specific sentences.

Attention based on Visual Regions Due to the drawbacks of global CNN features, most efforts focused on extracting more relationships from visual content. Applying the attention mechanism was a significant inspiration. The attention can be summarised to weighted averaging [88]. Based on attention, several works further improved the performance of image captioning with the bottom-up and top-down mechanisms. The bottom-up path executes the visual feedback to adjust previous predictions, while top-down indicates that leveraging prior information and inductive bias predict the upcoming sensory input [76]. Specifically, one representation of the top-down method means additive attention. Regarding this method, the language model attends a feature grid while predicting the next word [89]. Anderson et al. [90] defined the bottom-up through the object detector providing image regions. Then, a top-down method is combined with it to learn weighing each region for each word generation. Faster R-CNN [91] provides the pooled feature vectors of the region proposals from detected objects based on pre-training on Visual Genome [92] in

this approach. Benefits from image region feature, many of the subsequent research efforts have widely applied the visual encoding approach from this strategy such as [93] [94].

Graph-based Encoding Furthermore, some studies used graphs to establish image regions improving the representation of image regions. First, Yao et al. [95] and Guo et al. [96] integrated spatial and semantic information from objects by graph convolutional network (GCN) [97]. Aiming at estimating semantic relations, Yang et al. [98] designed the graph-based representation of the image and sentence, integrating semantic knowledge of text into the image encoding. Meanwhile, Shi et al. [99] applied the same strategy to represent the image while using ground-truth captions train module of predicate nodes.

Self-attention Encoding For language understanding and machine translation tasks, self-attention was first proposed by Vaswani et al. [75], which promoted the creation of Transformer-based architectures improving the performance in the NLP area and computer vision. Yang et al. [100] first leveraged a self-attentive operator to extract the relationships between objects. After that, Li et al. [101] designed a Transformer model based on combining a visual encoder with a semantic encoder. The structures of the two encoders mainly consisted of self-attention and feed-forward layers. Many of the following efforts proposed the variants of the self-attention module regarding image captioning [102] [103] [104] [105]. Furthermore, He et al. [106] proposed a spatial graph transformer to integrate different types of spatial relationships between detected objects. The extension of the attention module was proposed by Huang et al. [107], which weights the final attention outputs through a gate guided by the context. Pan et al. [108] proposed X-liner Attention applying bilinear pooling to improve the representation of the attended output feature. Meantime, Cornia et al. [109] designed a Transformer-based model to augment each encoder layer with a set of memory vectors, namely Memory-augmented Attention. Luo et al. [110] combined grid and region features to complement advancements of each other. In the other line, some works directly applied on image patches with Transformer-based structure [72]. Based on this strategy, Liu et al. [111] first executed a convolution-free framework in image captioning. Specifically, the en-

coder of the framework was pre-trained Vision Transformer network (ViT) [72] and Transformer as decoder generated captions. Interestingly, CLIP [112] was trained from scratch on large-scale data by the same visual encoding methods. Especially, CLIP-based features have been widely applied in image captioning [113] [114]. Additionally, Zhang et al. [115] designed VinVl based on BERT-like architecture, which proposed a new object detector to extract better image features and the vision-and-language pre-training objectives.

2.3.2 Language Models

The language model is the most crucial sequential information in image captioning, which predicts the probability of each word occurring in a sentence. This thesis mainly reviews two language modelling directions for image captioning: 1. LSTM-based methods; 2. Transformer-based methods.

LSTM-based Model Language as typical sequential information, recurrent neural networks (RNNs) have been applied to solve image captioning. Among RNNs variants, LSTM [116] is the most predominant language model. As a simple LSTM-based captioning model, Vinyals et al. [77] designed the single-layer LSTM. In this architecture, the image features from visual encoding are executed in the initial hidden state of the LSTM, generating the caption word by word. During the testing period, the generated words at the previous step represent the input words, while during the training period, the words of ground-truth captions are the input. After that, Xu et al. [117] proposed LSTM with an attention strategy. In this line, the attention mechanism was guided by the previous hidden state over the image features, which computes a context vector to feed into a multilayer perceptron (MLP) to generate output words. Many of the following modifications applied a decoder based on the single-layer LSTM. For example, Lu et al. [118] designed a visual sentinel, which is a learnable vector to augment visual features. When the decoder generates the "non-visual" words, the visual sentinel replaces the visual features. Meantime, Chen et al. [119] applied a second LSTM to reconstruct the previous state based on the current state to enhance the language model. Then, Wang et al. [120] decomposed the sentence generation into two periods for generating a sentence from coarse to

finer based on the single-layer LSTM, which were skeleton caption generation and attributes enriching. Meanwhile, Gu et al. [121] applied a sequence of LSTM decoders to design a coarse-to-fine structure generating refined sentences. Based on the performance of the single-layer LSTM, a two-layer LSTM was first proposed by Donahue et al. [79] to apply for the captioning model, which stacks two layers, where the input of the second layer comes from the hidden state of the first layer. Furthermore, Anderson et al. [90] exploited the attention mechanism in the two-layer LSTM. Due to the advancement of two-layer LSTM, several efforts designed variants aiming at two-layer LSTM. For instance, Lu et al. [122] proposed a pointing network guiding the content-based attention strategy. Remarkably, the networks predict some slots in the caption generation process, which are filled into the visual region classes. Huang et al. [107] used the Attention on Attention method to boost the LSTM, computing another step of attention on visual self-attention. Pan et al. [108] enhanced the language model and visual encoding through the proposed X-Liner attention mechanism.

Transformer-based Model Transformer has completely obtained breakthroughs in NLP, which were also applied in the image captioning task. The decoder of the Transformer uses the masked self-attention and cross-attention operation to apply to words, where words are the queries, and the keys and values are represented from the outputs of the last encoder layer with a feed-forward network. Regarding the advancement of the Transformer, some image captioning works have widely applied the standard Transformer decoder [123] [124] [110] [125]. Based on the variants of the Transformer model, Li et al. [101] modified the cross-attention operator by proposing a gating mechanism, which constrains the visual and semantic information by integrating image regions and semantic attributes from the external tagger. Similarly, Ji et al. [126] proposed the context gating strategy to adjust the effecting of the visual representation on each predicted word by multi-head attention. Cornia et al. [109] proposed modulating all encoding layers instead of the last one of cross-attention. Specifically, the proposed meshed decoder integrates the contributions of all the encoding layers; then, the weights of these contributions were guided by the semantic query.

CHAPTER 3

Sequential Control Process with Reinforcement Learning

In this thesis, we first investigate the sequential information control problem for process control. As with most conventional control problems, the traditional control theory has widely optimised the batch process. In addition, deep neural networks, as an advanced artificial intelligence technology, are also used in the batch process. In this chapter, we focus on process control by reinforcement learning.

Sequential control processes involve making a series of decisions over time to achieve a specific goal. These types of processes are common in many areas, such as robotics, finance, and manufacturing. Reinforcement learning, on the other hand, is a type of machine learning that involves an agent learning to make decisions based on the feedback it receives from the environment. When these two concepts are combined, the result is a powerful tool for optimising sequential control processes. Reinforcement learning allows the agent to learn from its actions and adjust its decisions over time, leading to better performance and more efficient outcomes. This is particularly important in complex systems where it may be difficult or impossible to determine the optimal sequence of actions through traditional methods. For another example, batch process control involves the manufacturing of a specific quantity of a product in a single batch. The process involves a series of steps, such

as mixing, heating, cooling, and chemical reactions, that must be performed in a specific sequence and with precise control of various parameters.

In this thesis, our study on sequential control process with reinforcement learning is based on the batch processes which are important manufacturing routes for the agile manufacturing of high-value-added products and they are typically difficult to control due to highly nonlinear characteristics, unknown disturbance and model plant mismatches.

3.1 Introduction

Batch processes have received more and more attention as they are suitable for the responsive manufacturing of high-value added products such as special chemicals, special polymers and pharmaceuticals [127]. Different with continuous process, batch process aims to produce chemicals in limited quantities at a fixed time frame [40]. Meanwhile, batch processing is highly adaptable and allows for flexibility in production. This flexibility is beneficial in industries where product variations, customization, and changing production requirements are common. It also enables the production of multiple products within the same facility [7]. The chemical batch process consists of a sequence of steps that are executed in a fixed order. These steps can include raw material addition, cooling, heating, mixing, filtration, chemical reactions, separation, and more. Each step is carefully controlled to ensure product quality and consistency [128].

With the increasing popularity of customized or personalized chemical and medicine, fed-batch and batch processes have become the major means of responsive manufacturing. Generally, the end-of-batch product quality is the main interest in fed-batch and batch process operations [128] [129]. It is well known that batch processes, such as batch polymerization reactors, are very difficult to be optimized and controlled due to issues associated with batch processes such as time-varying characteristics, non-steady operations and non-linearity. Besides, batch process can be less efficient than continuous processes for large-scale production. To address this, efforts are made to optimize batch processes, reduce cycle times, and minimize waste. This

may involve process automation and advanced control systems [128].

The goal is to produce a high-quality product that meets the desired specifications. Reinforcement learning can be used to optimise the batch process control by allowing the control system to learn from past batches and adjust the control parameters in real-time to produce a better product. The reinforcement learning agent receives feedback from the system based on the quality of the product produced and adjusts the control parameters accordingly. For example, consider the production of a batch of pharmaceuticals. The process involves several steps, such as mixing, heating, cooling, and chemical reactions, and the quality of the final product depends on several parameters, such as the temperature, pressure, and duration of each step. A reinforcement learning agent can be trained to optimise the control of the process by learning from past batches. The agent can receive feedback from the system based on the quality of the final product, such as the purity and yield of the pharmaceuticals. Based on this feedback, the agent can adjust the control parameters, such as the temperature and pressure, to improve the quality of the next batch. Over time, the reinforcement learning agent can learn to optimise batch process control and produce high-quality pharmaceuticals consistently. This can lead to significant cost savings and increased efficiency in the manufacturing process.

3.2 Related Work

In order to overcome these difficulties, a number of advanced data-driven modelling techniques such as neural networks and hybrid computational intelligence methods have been utilized in building models for batch processes [130]. Neuro-fuzzy system combined with an adaptive controller to control penicillin production was studied by Bravo [131]. Nagy proposed a neural network predictive control strategy to control a biochemical reactor [132]. A feed-forward neural network was applied to optimize and control batch processes [128].

Compared with the above control algorithms, reinforcement learning control can be used as an alternative optimization control strategy for the control of batch processes. Firstly, the optimal control signal can be obtained by choosing actions

after the online interaction between environment and active decision-making agent. Secondly, the previous process information is utilized by controller design. Finally, due to the fact that this is a simple control algorithm, low-cost hardware can be used. Hence, reinforcement learning control has been applied to various sophisticated problems or devices in recent years. For example, Sutton and Barto described that the control targets could be responded to from the value of the reinforcement at each time, which includes errors, profits or cost [40]. The problem of adaptive control of a nonlinear chemical process was solved by applying Q-learning by Shah and Gopal [133]. Said and Guido applied Q-learning to control and optimize the operation of robots [40]. The path length of Nanobots was optimized by using Q-learning by Lambe [133] and Spielberg et al. [40] used Q-learning for process control.

Although reinforcement learning has been applied in many areas, its reported application to batch processes is limited. Therefore, in order to overcome the difficulties in batch process optimization and control, the stochastic multi-step action Q-learning algorithm and the modified multiple step action Q-learning are proposed for the optimization control of fed-batch process operations in this paper.

3.3 Methodology

3.3.1 Modified Multiple Step Action Q-learning algorithm

Firstly, we propose the Modified Multiple Step Action Q-learning algorithm (MMSA) based on MSA. Compared with traditional Q-learning and MSA, MMSA applies an important modified $\varepsilon - greedy$ policy. The modification is that the agent follows different ε probability to explore and select actions: the agent follows a suitable ε value to explore actions during most of the training time in which the agent can obtain optimal actions. After that, the ε value can be changed to small, making the agent explores and chooses non-optimal actions as less as possible. Due to applying this modified $\varepsilon - greedy$ policy, more and more optimal actions and rewards are saved into the Q-table to improve the efficiency of exploration and learning.

3.3.2 Stochastic Multi-step Action Q-learning Algorithm

Second, we further propose the Stochastic Multi-Step Action Q Learning Algorithm (SMSA) to improve MMSA. Compared with MMSA, the modification is that the agent can not apply the same action in multi-steps within a fixed time, but it can execute the same action in multi-steps within a random different time. Regarding this modification, the agent cannot compulsively and persistently select the same non-optimal actions in many steps. The exploration of action is flexible, and the agent can rapidly learn optimal actions to acquire more and more goal states. The updating of the Q-value function is modified in SMSA as:

$$Q(s_i, a_i^{n_k}) \leftarrow Q(s_i, a_i^{n_k} + \alpha_t \left[R + \gamma \max Q(s_{i+1}, a_i^{n_k}) - Q(s_i, a_i^{n_k})\right] \tag{3.1}$$

Here, s_i is state, $a_i^{n_k}$ indicates an action a_i is applied in n time steps in the kth period, $Q(s_i, a_i^{n_k})$ is Q-value and $\max Q(s_{i+1}, a_i^{n_k})$ is the maximum Q-value of next new state. The whole system running time will be divided into different n time steps of k parts. For example, initially, if time step $n_1 = 5$, the agent will choose same action a_1 from state s_1 to s_5. But if next time step $n_2 = 3$ which means that s_6, s_7 and s_8 will execute the same next action a_2.

The reinforcement learning model is different from traditional machine learning in the process control; it follows the specific process to design the agent, state, action and reward function. To prove the control performances of MMSA and SMSA, we apply them to the Case 1 and Case 2 processes in the experiments section. Hence, the experiments section replaces the methodology section to describe the design of the agent, state, action and reward function.

3.4 Experiments

Most batch processes have two main goals. The first is that the the reaction within the limitation time can produce more desired products. The second is that the control strategy is smooth, which is more stable in the real-world. Based on these two goals, the final values of desired product C are the main evaluation metric on

both cases. Differently, in case 2, the goal of the control strategy is to produce more desired products while ensuring fewer undesired products are produced.

Furthermore, compared with typical machine learning scenarios, the dataset and environment are not similar. In these experiments, the control signal must interact with environment to obtain the control results. In both cases, the simulation functions serve as the representation of the environment.

3.4.1 The application of MMSA

Case 1

The nonlinear batch reactor process used by Xiong and Zhang [128] is taken as a case study. In this case study, the control objective is to maximize the amount of desired product at the end of a batch. The reaction in this batch process is described by the following equation.

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C \tag{3.2}$$

In this reaction, the raw material is reactant A, B is the desired product, and C is the undesired by-product. With the control of reactor temperature, the desired product B will be generated in a specified batch time $t_f = 60min$. Based on material balances and reaction kinetics, the following mechanistic model can be developed:

$$\begin{aligned} \frac{dx_1}{dt} &= -k_1 exp(-E_1/uT_{ref})x_1^2, \\ \frac{dx_2}{dt} &= -k_1 exp(-E_1/uT_{ref})x_1^2 - k_2 exp(-E_2/uT_{ref})x_2, \\ u &= T/T_{ref}, \end{aligned} \tag{3.3}$$

where the concentrations of A and B are represented by x_1 and x_2 respectively; the reference temperature is represented by T_{ref}.

The Table 3.1 gives the values of k_1, k_2, E_1, E_2 and T_{ref}. The initial conditions of A and B are $A(0) = 1$ and $B(0) = 0$. In addition, the range of reactor temperature is $298K \leq T \leq 398K$. Based on this detailed mechanistic model, a simulation of this batch process is developed using MATLAB, and the simulation is used to test various control strategies.

Table 3.1: Parameter for batch process of Case 1

Parameter	***Value***
k_1	4.0×10^3
k_2	6.2×10^5
E_1	2.5×10^3
E_2	5×10^3
T_{ref}	$348K$

Table 3.2: Parameters for MMSA

Variable	***Meaning***	***Setting***
α	learning rate	0.09
γ	discount factor	0.87
ε(0-34min)	greedy-probability	0.6
ε(34-60min)	greedy-probability	0.07

The Agent Design of MMSA

As an element of the MMSA algorithm, the model of the agent is important. For the agent, several parameters need to be designed. The discount factor γ and the learning rate α are vital elements, which can decide the speed of learning and control system model updating. For MMSA, the most important element is the implementation of $\varepsilon - greedy$ policy and the parameters are shown in Table 3.2:

The State Design of MMSA

In this paper, the main control goal is that the desired product should be maximized during a fixed reaction time. Hence, the state model is designed to follow this principle. Following the implementation of the control signals, the desired product can experience an increase. For this process, if the slope of desired product B curve is made big, the task goal can be achieved. Therefore, the state is represented by

Table 3.3: States of MMSA

Equation	***State***
$B(t+1) - B(t) \geq 0$	goal state
$B(t+1) - B(t) < 0$	undesired state

the derivative (first order difference) of the desired product B as shown in Table 3.3:

The action design of MMSA

The update and transition of the state are decided by the exploration and selection of action. Generally, the goal state can be reached after executing optimal action. With the change in temperature, the desired product B can be produced which means that the variation of temperature can influence the final concentration of desired product B. Therefore, the action model is designed by temperature T as the control signal in this case study. And there are 206 different symbolic actions. Figure 3.1 indicates the action space in this experiment.

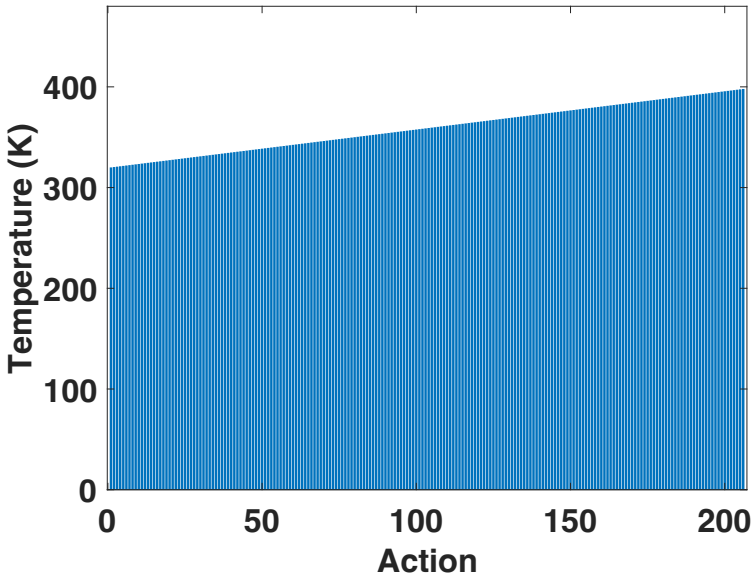


Figure 3.1: Action space of MMSA

The reward function of MMSA

In reinforcement learning, the learning control performance can be influenced by the rewards. Thus, the reward function is a key element for the MMSA algorithm. And some constant values can represent reward functions for reinforcement learning. However, the performance of using constant values as the reward function is not suitable. In this case, the reward function will be redesigned. The derivatives of desired product B describe the state meaning that the bigger this positive value is, the better control objective can be acquired. Due to this characteristic, the derivative of desired product B represents the reward function:

$$Reward = \Delta[B] \tag{3.4}$$

According to this dynamic reward function, the good and bad performance of exploration and selection of action can be directly and clearly reflected in each step.

Control Results and Discussions of Case 1

In this part, the batch process will be controlled and optimized by the MMSA algorithm. To demonstrate the advantages of MMSA, the control performance of MMSA will be compared with traditional Q-learning and MSA.

In this case study, the main control goal is to maximize the desired product B. Figure 3.2 shows the concentration profiles of the desired product B under different control algorithms in a fixed reaction time. At the end of the batch, the final values of the desired product B are 0.557, 0.587, and 0.640 under the traditional Q-learning, the MSA algorithm, and the MMSA algorithm respectively. Among the three control strategies, the MMSA algorithm gives the best control performance.

The control signals of different control algorithms are shown in Figure 3.3, Figure 3.12 and Figure 3.5. It is clear that the control signals of the MSA and MMSA are more reasonable compared with that of the traditional Q-learning. To acquire more rewards, the selection of action of traditional Q-learning is not limited to the fix $\varepsilon - greedy$ policy. Hence, the control signal of the traditional Q-learning exhibits a lot of sharp oscillation which is not reasonable in the real batch process.

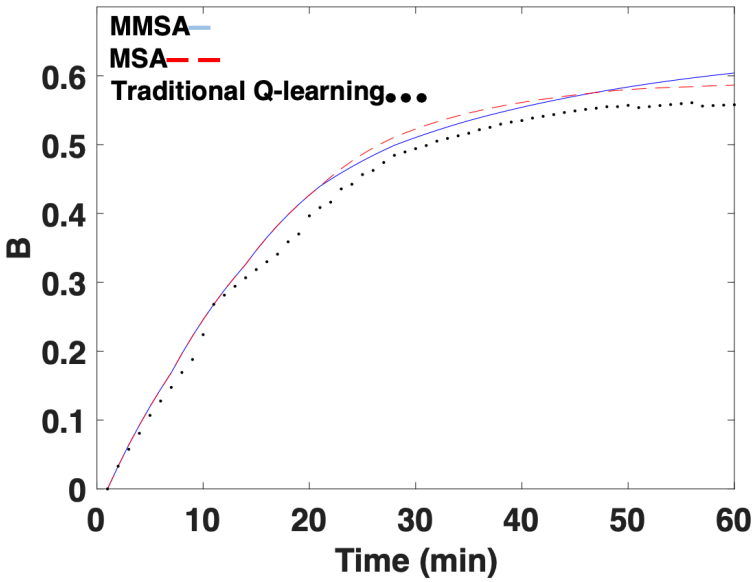


Figure 3.2: Comparison of final desired product for three algorithms

According to Figure3.4 and Figure3.5, the control signals of MSA and MMSA are clearly more reasonable. For MSA, Figure3.4 shows that the curve of the whole control signal experiences steady and persistent oscillation meaning that the control signal (action) of MSA is explored continuously.

In other words, the agent explores and selects optimal actions; meanwhile, some non-optimal actions are executed many times in the whole batch time, which means that a number of rewards will be received, while amounts of punishments will come along with them. The reason is that the agent persistently follows the fix $\varepsilon - greedy$ policy to explore and choose available actions resulting in the performance of Q-table and control efficiency. On the contrary, it is different for the MMSA algorithm. From Figure 3.5, the curve of the control signal can experience distinct oscillation and variation firstly, then it will be changed to smooth and steady. Since MMSA has the improved $\varepsilon - greedy$ policy, the agent can explore and select more and more available actions with big ε value during the first reaction time (0-34min); after that, there is a small ε value to explore actions (34-60min), which indicates that there are more and more optimal actions and rewards can be obtained compared with MSA algorithm. According to this modification and improvement, there is a better Q-table, while the exploration of action of MMSA has a higher efficiency. Therefore, when the MMSA algorithm controls this batch process, there will be better control

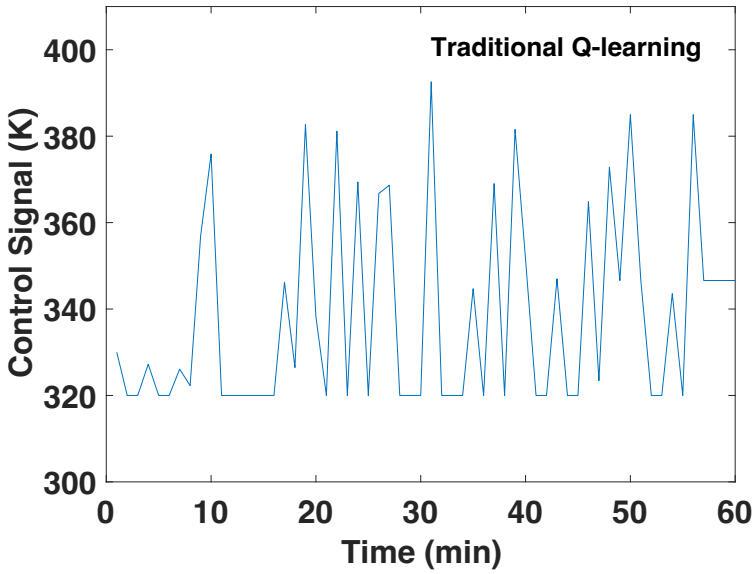


Figure 3.3: Control signal of traditional Q-learning

efficiency and robustness.

Besides what has been mentioned above, the reward distributions of different algorithms can indicate the advantage of MMSA which can be discussed in Figure3.6, Figure3.7 and Figure3.8. Generally, it is common that rewards and punishments are the most important and significant elements which decide the good and bad of the Q-table directly leading to the performance of reinforcement learning. According to Figure3.6 and Figure3.7, it is obvious that there are a number of punishments saved into the Q-table when traditional Q-learning and MSA control batch process resulting in bad influence for reinforcement learning that Q-table cannot instruct better the agent to choose more optimal actions.

However, Figure 3.8 indicates that when MMSA control this batch process, more and more rewards can be acquired without punishments in the same fixed reaction time. Hence, the Q-table can instruct the agent to explore and choose actions better which means that the control task can be achieved fast and accurately. This comparison can demonstrate that the MMSA algorithm can control the batch processes better.

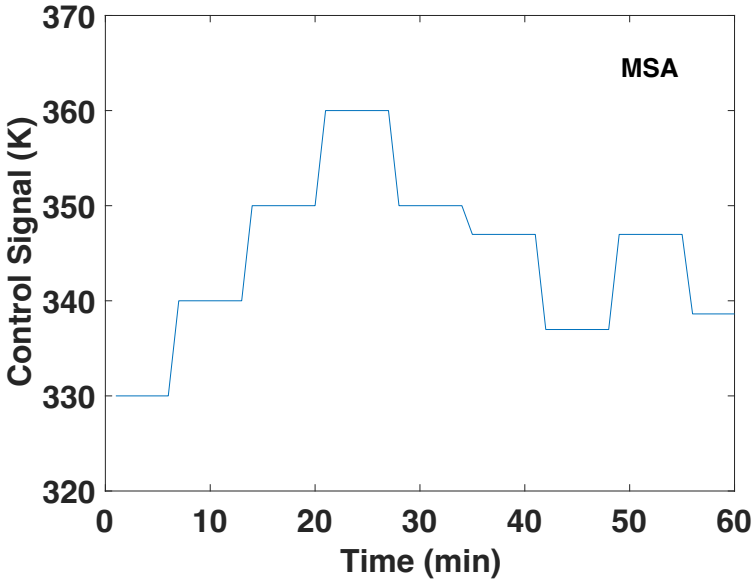


Figure 3.4: Control signal of MSA

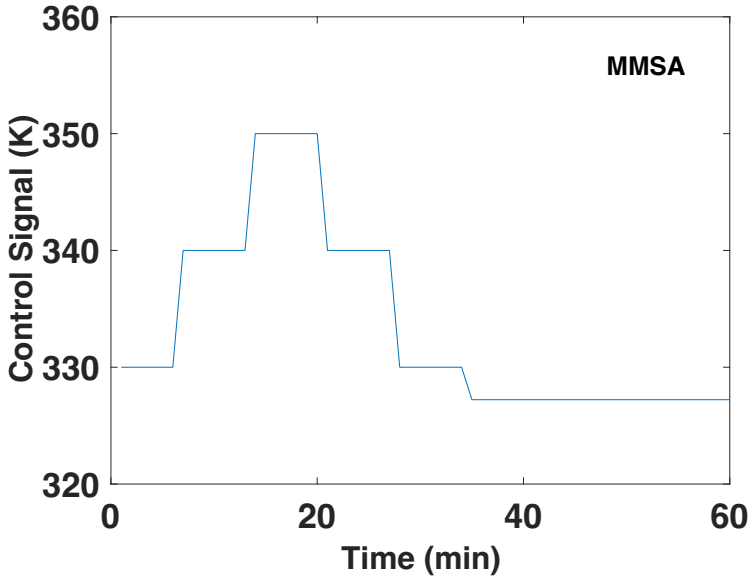


Figure 3.5: Control signal of MMSA

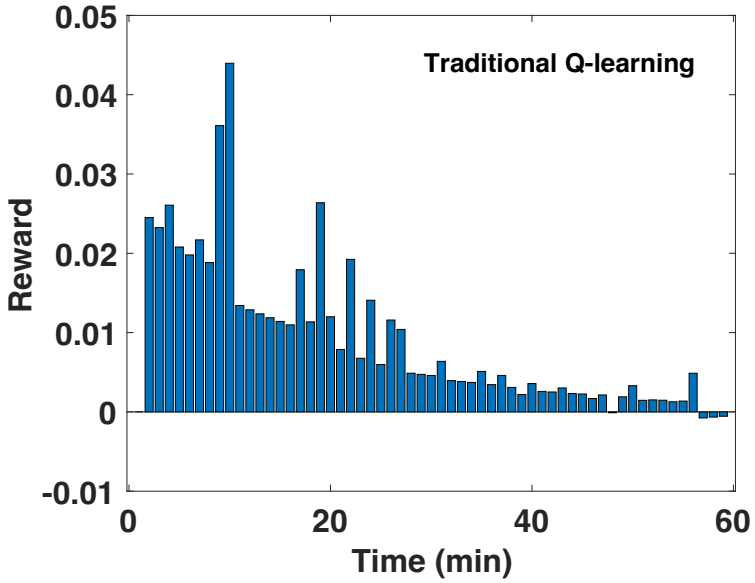


Figure 3.6: Reward distribution of traditional Q-learning

3.4.2 The application of SMSA

Case 2

A fed-batch process [127] is used as a case study. The following equations describe the reactions in this fed-batch process.

$$A + B \xrightarrow{k_1} C, \tag{3.5}$$

$$B + B \xrightarrow{k_2} D, \tag{3.6}$$

For this reaction system, reactants A and B are the raw materials and the desired product is C and the species D is the undesired by-product. In a specified batch time $t_f = 120min$, the reactant B will be added into the reactor gradually to prevent the fast formation of the undesired by-product D. Therefore, the main control objective is that the desired product C should be acquired as much as possible and the undesired species D should be kept at the lowest quantity. The concentration of reactant B in the feed is $b_{feed} = 0.2$. Based on material balances and reaction kinetics, the following mechanistic model can be developed:

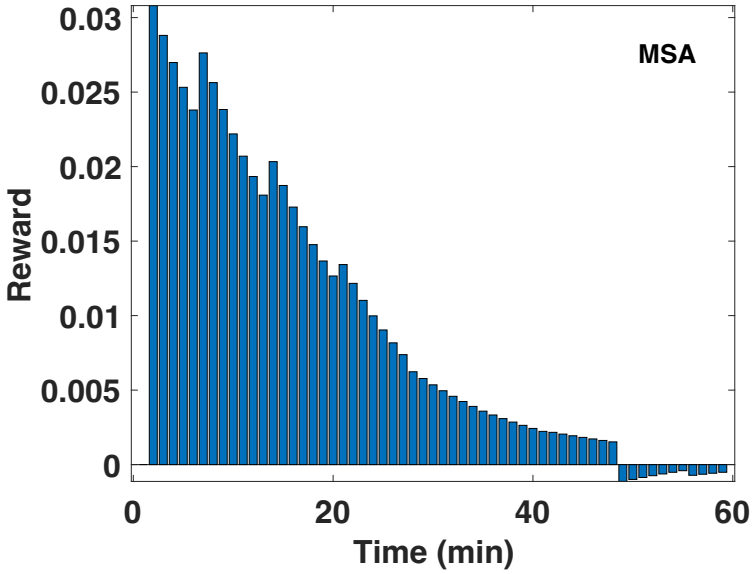


Figure 3.7: Reward distribution of MSA

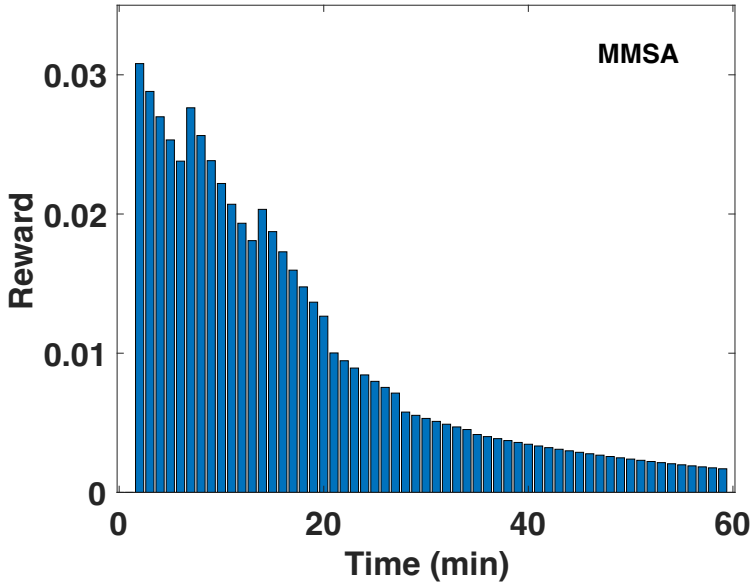


Figure 3.8: Reward distribution of MMSA

$$
\begin{aligned}
\frac{d[A]}{dt} &= -k_1[A][B] - \frac{[A]}{V}u, \\
\frac{d[B]}{dt} &= -k_1[A][B] - 2k_2[B]^2 + \frac{b_{feed} - [B]}{V}u, \\
\frac{d[C]}{dt} &= -k_1[A][B] - \frac{[C]}{V}u, \\
\frac{d[D]}{dt} &= 2k_2[B]^2 - \frac{[D]}{V}u, \\
\frac{d[V]}{dt} &= u.
\end{aligned} \tag{3.7}
$$

The concentrations of A, B, C and D are represented by $[A]$, $[B]$, $[C]$ and $[D]$ respectively. The volume of the materials in the reactor and reactant feed rate is denoted by V and u respectively, and k_1 and k_2 are the reaction rate constants. Table 3.4 shows the parameter values. Based on this detailed mechanistic model, a simulation programme of this fed-batch process is developed using MATLAB and the simulation is used to test various control strategies.

Table 3.4: Parameters for the fed-batch process of Case 2

parameters	***value***
k_1	0.5
k_2	0.5
$[A](0)$	0.2moles/litter
$[B](0)$	0
$[C](0)$	0
$[D](0)$	0
$[V](0)$	0.5

The Agent of SMSA

As an element of the SMSA algorithm, the model of the agent is important. For the agent, several parameters need to be designed such as the discount factor γ and the learning rate α. The $\varepsilon - greedy$ policy is an essential algorithm for the SMSA algorithm. In the batch process, because plenty of positive influence will be

generated at the beginning of the process and the last period of the process has the negative influence, there is a bigger ε value in most of the whole batch time firstly, by contrary, a smaller ε value will be applied. In this study, the parameters are determined based on experiments and are shown in Table 3.5.

Table 3.5: Parameters used in the simulations of Case 2

variable	***meaning***	***setting***
α	learning rate	0.1
γ	discount factor	0.98
ε(0-79min)	greedy-probability	0.8
ε(79-120min)	greedy-probability	0.05

The State of SMSA

In this process, the main control goal is that the desired product should be produced as much as possible and the undesired by-product should be kept at a low quantity at the final time. In this study, the state model is designed to follow this principle. During a given reaction time, each product can experience an increase or decrease following the implementation of the control signals. In this process, when the rate of increase of the desired product concentration is high, the more desired product is expected to be produced. At the same time, the rate of increase of the undesired product curve should be kept slow. According to this principle, making the slope of the $[C]$ curve bigger than that of the $[D]$ curve can achieve the control task in the whole batch time. Therefore, the state is represented by the difference in derivatives between $[C]$ and $[D]$ as shown in Table 3.6, where $\Delta[C]$ and $\Delta[D]$ are the rates of change in $[C]$ and $[D]$ respectively.

The Action of SMSA

Generally, the exploration and selection of action decide the update of state, which indicates that when the agent explores and chooses an optimal action, the goal state

Table 3.6: States of Case 2

Condition	***State***
$\Delta[C] - \Delta[D] \geq 0$	goal state
$\Delta[C] - \Delta[D] < 0$	undesired state

will be reached. Thus, the action is a significant element of this SMSA algorithm. In this study, the addition rate of reactant $[B]$ will influence the final quantity of product $[C]$ and by-product $[D]$. Considering this, the action model is the feeding rate of reactant $[B]$, the control signal u ranges from 0.0020 to 0.01. For this fed-batch process, the actions can be explored and selected randomly from this range to significantly reduce the bias and errors with improved accuracy.

The Reward Function of SMSA

The reward function is also a significant key element in the SMSA algorithm, due to the fact that the rewards and punishments of the reward function can significantly influence the learning control performance. For the conventional reinforcement learning algorithm, it is common that the reward function can be set as some constant values. However, using constant values as the reward function is not suitable and reasonable for the batch process. Therefore, the reward function is redesigned in this fed-batch process. In this case, the difference in derivatives between desired product $[C]$ and undesired by-product $[D]$ describes the state. In other words, the bigger this positive value difference is, the better the control objective can be obtained. Due to this characteristic, the reward function can be described by the cubic difference in rates of change between desired product $[C]$ and undesired product $[D]$:

$$Reward = (\Delta[C] - \Delta[D])^3 \tag{3.8}$$

This dynamic reward function can directly and clearly reflect the good and bad performance of exploration and selection of action in this step.

Control Results and Discussions of Case 2

The performance of controlling this fed-batch process using SMSA is tested on simulation. In order to demonstrate the advantages of SMSA, its results are compared with those of other control algorithms including traditional Q-learning, multi-step Q-learning (MSA) and ne uro-fuzzy networks-based optimization control.

The variations of desired product $[C]$ and undesired product $[D]$ of different control algorithms are shown in Figure 3.9 and Figure 3.10 respectively. The end of batch values of the desired product $[C]$ and the undesired product $[D]$ are given in Table 3.7.

Table 3.7: The comparisons with the state-of-the-art algorithms on Case 2

Algorithm	***Desired product*** $[C]$	***Undesired product*** $[D]$	***Final difference of*** $[C]$ ***and*** $[D]$
Neuro-fuzzy network	0.0559	0.0204	0.0319
MSA	0.0585	0.0227	0.0358
Traditional Q-learning	0.0590	0.0193	0.0397
SMSA	0.0618	0.0249	0.0369

Based on Figure 3.9 and Table 3.7, it is clear that by applying SMSA control to this process, plenty of the desired product $[C]$ can be generated, more than those from the other control algorithms. Although when applying the traditional Q-learning, MSA and neuro-fuzzy networks, less final quality of undesired production $[D]$ are produced as shown in Figure 3.10, the difference between desired product $[C]$ and undesired by-product $[D]$ under SMSA are much larger than those under MSA and neuro-fuzzy network as indicated in Table 3.7, which demonstrates that SMSA can be considered as giving better control performance than MSA and neuro-fuzzy network.

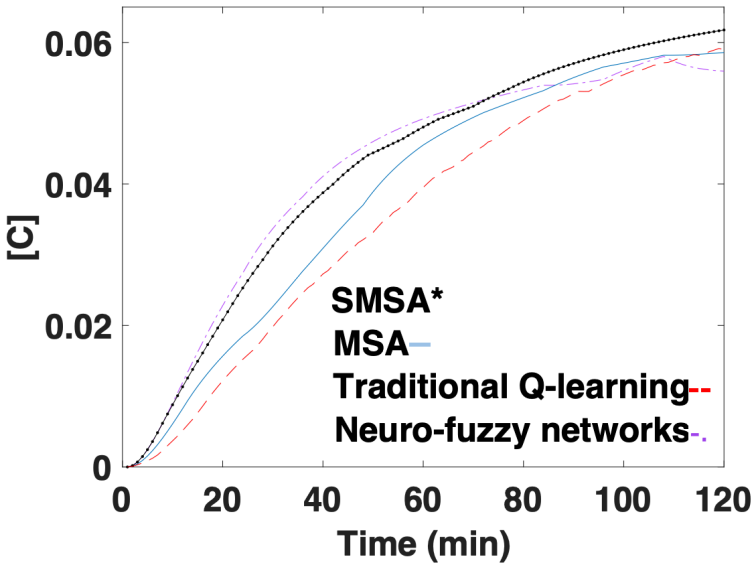


Figure 3.9: Variation of [C] in different algorithm

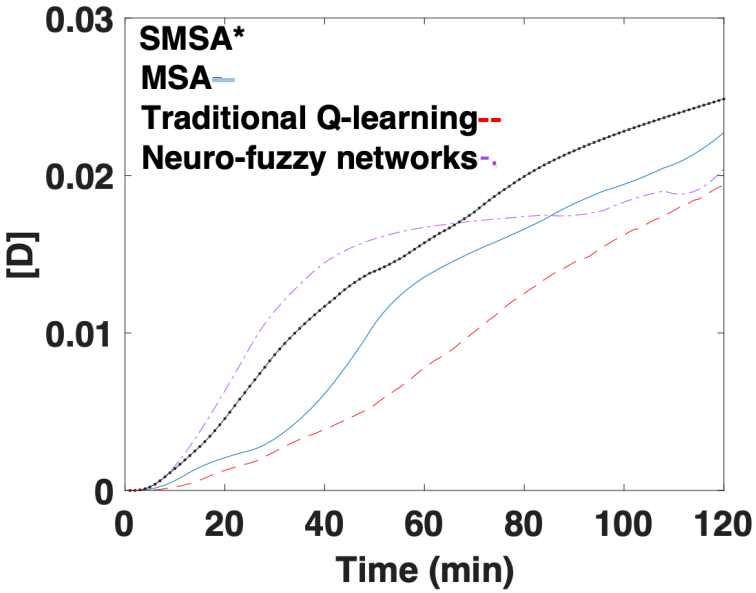


Figure 3.10: Variation of [D] in different algorithm

According to Figure 3.9, Figure 3.10 and Table 3.7, the traditional Q-learning and MSA have a good control performance as well, in particular, the traditional Q-learning control results in the largest difference between the desired product $[C]$ and the undesired by-product $[D]$. However, the control policies under the traditional Q-learning and MSA are not optimal and reasonable compared to SMSA on the final volume and control signal.

At the end of the batch, the final volume are $0.99m^3$, $1.24m^3$, $0.92m^3$ and $0.98m^3$ when applying neuro-fuzzy network-based optimization control, MSA, traditional Q-learning and SMSA. Considering the maximum volume constraint of $1m^3$, this batch process experiences enough reaction by the neuro-fuzzy network and the volume is too large being not reasonable in the industries applying MSA. Compared with SMSA, the traditional Q-learning leads to less final volume. Hence, there is no doubt that when SMSA control is applied, more desired production $[C]$ and less undesired production $[D]$ can be acquired the final volume being close to its constraint.

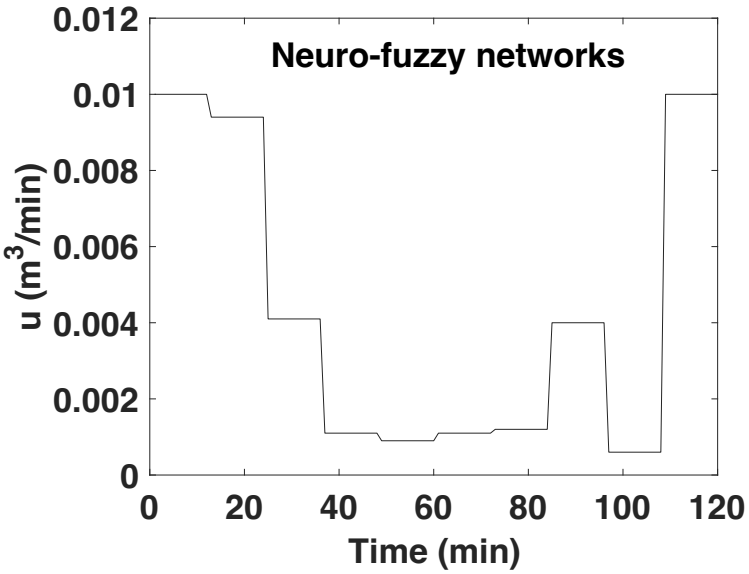


Figure 3.11: Control signal of neuro-fuzzy networks

Figure 3.11, Figure 3.12, Figure 3.13 and Figure 3.14give the control signals of the different control algorithms. It is obvious that the control signals of SMSA, MSA and neuro-fuzzy network-based optimization control are more reasonable. As a learning algorithm, the advantage is that SMSA, MSA and traditional Q-learning will learn first, then apply learned optimal policy control task, which means that

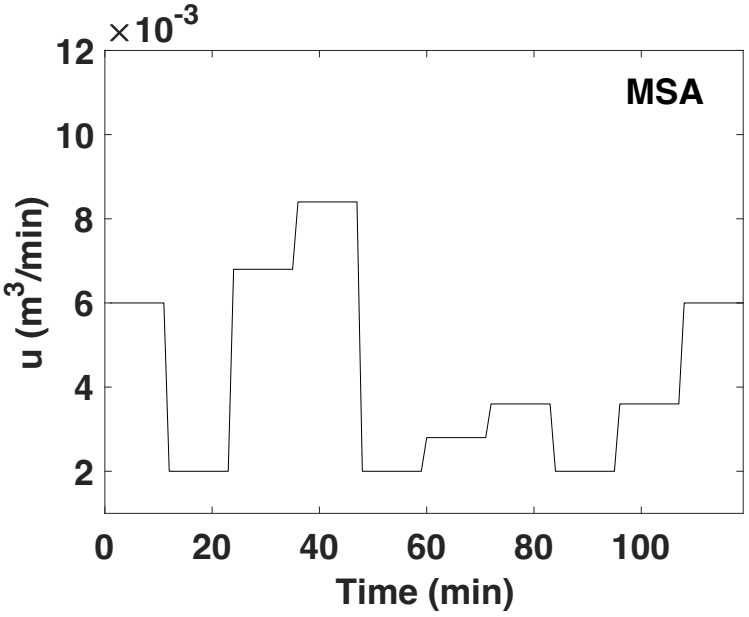


Figure 3.12: Control signal of MSA

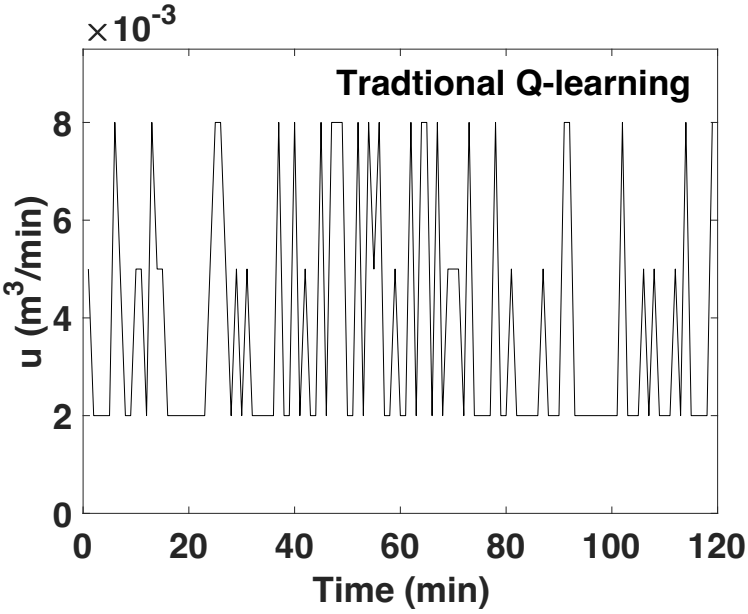


Figure 3.13: Control signal of traditional Q-learning

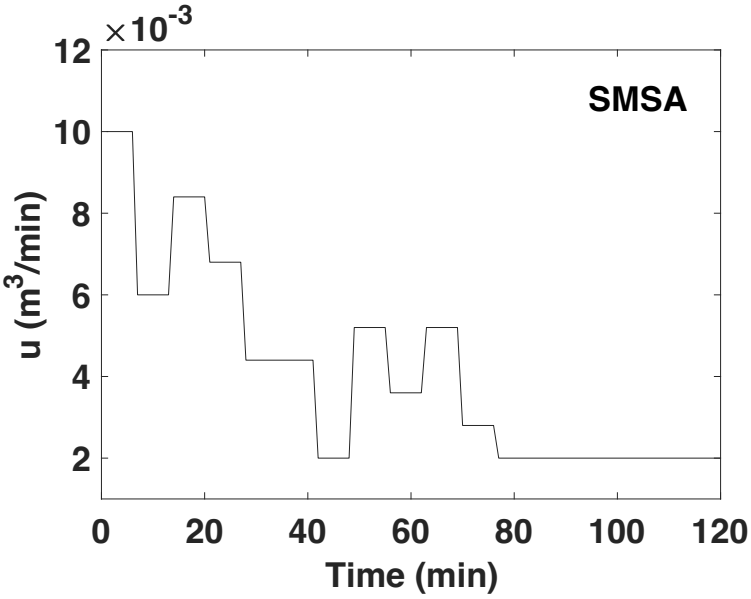


Figure 3.14: Control signal of SMSA

they will explore available actions to reach the next new state and receive a reward or punishment.

However, the traditional Q-learning will select and explore a new action to reach a new state in each sampling time. Hence, the learned control signal of traditional Q-learning can experience sharp and frequent variations at each sampling time making the practical implementation in real industrial processes problematic. In contrast, the learned control policies of MSA and SMSA algorithms do not have such sharp variations indicating that MSA and SMSA are more reliable and robust. Therefore, the proposed SMSA algorithm is expected to give better performance for real industrial applications.

3.5 Conclusions

In the paper, the use of reinforcement learning (RL) for the control of a fed-batch process has been established, using the proposed MMSA and SMSA (successive mean-based search algorithm) algorithm. The goal was to optimise the end-of-batch production objectives in a highly nonlinear batch process without needing a process model. The results showed that the MMSA and SMSA algorithm outperformed MSA (mean-based search algorithm), traditional Q-learning, and neuro-

fuzzy network-based optimization control in terms of control performance, reliability, and robustness of the control policy. Compared with two proposed methods, the SMSA algorithm was found to have strong learning abilities and could rapidly optimise the end-of-batch production objectives. The proposed method was also able to acquire the optimal control policy in a short iteration time.

One possible future work would be to investigate the use of the proposed SMSA algorithm in different batch processes and compare its performance with other RL algorithms. We also consider applying the proposed method to real-world batch processes to evaluate its effectiveness in practical applications. The shortcoming of the study is that it is based on simulation results rather than real-world data. Therefore, the proposed method needs to be validated with experimental data to confirm its effectiveness in real-world batch processes. Additionally, the work did not compare their proposed method with other optimisation methods, such as model predictive control (MPC), which is a popular control strategy in the process industry. A comparison with MPC would provide insight into the advantages and disadvantages of both methods for batch process control. We address these problems in the next chapter. A flexible simulator-free framework is provided using neural digital twins. The neural predictor also provides guidance for reinforcement learning training as that in MPC.

CHAPTER 4

Hybrid Reinforcement Virtual Learning for Sequential Control

The advancements of reinforcement learning in the batch process have been proved in chapter 3. Hence, we further investigate the sequential information of process control aiming at control efficiency based on a hybrid reinforcement learning control strategy. Reinforcement Learning (RL)-based control system has received considerable attention in recent decades. However, in many real-world problems, such as Batch Process Control, the environment is uncertain, which requires expensive interaction to acquire the state and reward values. In this chapter, we present a cost-efficient framework, such that the RL model can evolve for itself in a Virtual Space using the predictive models with only historical data. The proposed framework enables a step-by-step RL model to predict the future state and select optimal actions for long-sight decisions. Under the experimental settings of the Fed-Batch Process, our method consistently outperforms the existing state-of-the-art methods.

4.1 Introduction

Batch processes, as an important chemical process, are expected to generate higher value products, such as desirable chemicals, polymers and pharmaceuticals [134],

which have received considerable attention in recent years. Due to the rapid evolution of diversely customised chemical processes, fed-batch is then considered to be one of the most popular approaches to responsive manufacturing. Among the fed-batch and batch process operations, the maximum end-of-batch product quality is the most noteworthy [134]. Batch processes usually face a dilemma in optimisation and control treatment, due to the rapid time-varying characteristics, non-steady operations and non-linearity batch polymerisation reactors [135].

The existing solutions are sought from *Modern Control Theory*, which experienced a rapid improvement in their optimisation mechanism. A number of optimal control approaches, *e.g.*, Proportional-Integral (PI), Proportional–Integral–Derivative (PID) and fuzzy control, have been applied in various disciplines. For example, Khalili *et al.* [136] proposed an optimal sliding mode control in biology. Trajectory optimization was then presented and applied in robotics by Carius *et al.* [137]. Wei *et al.* [138] applied such an optimal control to operate and optimize motor.

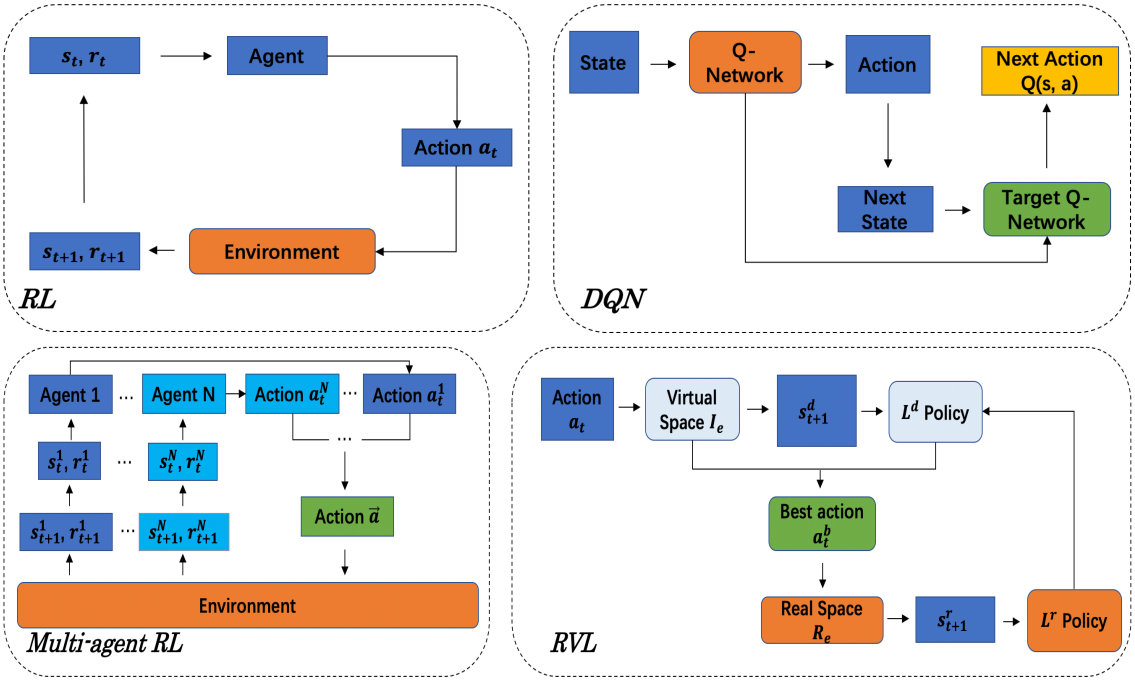


Figure 4.1: Several widely used methods of reinforcement learning, such as the traditional RL, DQN and multi-agent reinforcement. The differences between the proposed RVL and the existing models are highlighted.

With the rapid development of *Machine Learning* technologies, an emerging trend of modern control systems has been introduced by exploring the advanced

data-driven strategies, *e.g.*, neural networks and hybrid computational intelligence algorithms [134]. Particularly, *Reinforcement Learning* (RL) models have manifested the application values in many fields, such as computer vision [38] [139], games [140] and medicine [141]. With the development of neural networks, Deep Learning (DL) and RL models have been successfully applied in various chemical processes. For example, Jie *et al.* [127] applied the recurrent neural fuzzy network in the fed-batch process. Shah and Gopal [133] applied Q-learning to solve the problem of adaptive control of a nonlinear chemical process. Although RL has been applied in different chemical processes, it still lacks exploration in fed-batch processes.

In this chapter, a new structure of Reinforcement Virtual Learning (RVL) is proposed to control and optimize the fed-batch process. The novelty can be summed up as that the virtual space is explored and cooperated with RL, which means that a virtual environment can be predicted and created by previous data and the RL agent can further interact with the virtual environment to learn. Specifically, a simple and conventional prediction model is explored combined with RL to generate a more effective and flexible method. We summarise our main contributions below:

- The learned agent of RL through interaction with the virtual environment can acquire a virtual learning policy. When the agent of RL interacts with the real environment, this virtual learning policy can introduce and modify the agent to learn a real learning policy. According to this cooperation, RL can control and optimise the process in an uncertain environment. Also, previous historical information can be utilised adequately.

- Besides previous historical information, the proposed RVL can leverage future information as well. In terms of the virtual environment and learning policy, the results of future approximation control can be obtained. Considering the ability of future prediction, the results of discretionary future approximation control can then be acquired. The agent modifies and improves the learning policy based on the combination of the short-sight and long-sight approximations of the future. Hence, the previous historical information combined with future information can increase learning efficiency.

- The comprehensive experiments demonstrate that the results of control obtained by the proposed RVL show better performances compared with the state-of-the-art control algorithms for the fed-batch process.

The proposed structure of Reinforcement Virtual Learning (RVL) aims to address the time-series issues in batch-process control. In batch processes, the volume and capacity of reactants in the previous sequences significantly affect future reactions. RVL achieves this by extrapolating future information within a virtual space to provide rewards and penalties, optimizing the control strategy of previous sequences. The methodology section will provide more in-depth details

4.2 Related Work

As a conventional treatment of chemical process, the fed-batch process brings in high-profile exploitation, while the product costs and desired product quantity are the major control challenges. To solve this problem, a better control policy is expected. With the development of modern technology, control and optimization methods started to be applied in the fed-batch process in recent years. For instance, many theoretical works paid attention to step profiles to resolve the optimization issues for the fed-batch process [142] [143] [144]. Generally, the piecewise parameterization by the mean of linear polynomials is another kind of approach [145] [146]. The convenience of using such a smooth continuous feeding profile was marked by Martinez *et al.* [147]. The feed rates were parameterized by the sinusoidal functions developed by Ochoa [148]. The predictive control was also applied to control and optimize the fed-batch process [149] [150] [151]. However, the online determination and control of processing variables are not straightforward in the initial stage. After a period of development, it is still inefficient considering that there are plenty of works to take and analyze the samples. The reversibility and uncertainty of the processing models can influence the control performances and implementations in the real world.

With the development of machine learning/deep learning [38, 152, 153], there is plenty of research focusing on finding an alternative method to replace the tradi-

tional optimal control approaches. As a model-free algorithm of machine learning, RL was noticed and experienced rapid development in control area. The agent can find an optimal learning policy by a state-action value function based on the classic Q-learning [154]. To increase the efficiency of RL, Hausman *et al.* [155], Florensa *et al.* [156] and Kearns *et al.* [157] explored the latent models. In addition, Gupta *et al.* [158] applied the gradient-based fast adaptation algorithm to acquire exploration policy through using prior information. Garcia *et al.* [159] applied the meta strategy into Markov decision process (MDP) to obtain an optimal exploration strategy. Later, several kinds of methods combined with RL were proposed to further improve the overall performances. Mnih *et al.* [44] proposed a Deep Q-network (DQN) to estimate the state-action value function. Double DQN was then estimated [160] based on DQN to solve the problem of over-estimation of previous Q-network. After that, the state value and advantage value were predicted through the separated Q-network from Dueling Network explored by Wang *et al.* [161]. The strength of DQN was combined with constrained optimisation approach by the Optimally Tightening method by He *et al.* [162]. Harutyunyan *et al.* [163] and Munos *et al.* [164] combined on-policy samples into off-policy learning targets by $Q^*(\lambda)$ and Retrace(λ). Fortunato *et al.* [165] proposed a Noisy-Net to increase the ability of exploration by adding noise into the parametric model during the learning progress. Distributional RL [166] learned a value function using full distribution instead of expected values. Pritzel *et al.* [167] proposed a neural episodic control to generate semi-tabular representation and retrieve fast-updating values by context-based lookup for action selection. Lin *et al.* [168] improved the performance of DQN and proposed an episodic memory deep Q-network by distilling information of the episodic memory. Despite the success, these methods still need to combine different algorithms with RL, and thus, DQN relied on the open environment which only considers the prior experience without future information. In addition, treating the neural networks as a state-action value function cannot leverage future information to guide the learning of RL agent.

As one of the most important algorithms in multi-agent system, multi-agent reinforcement learning (MARL) gained traction recently with various successful ap-

plications. For example, Littman [169] studied MARL in the context of Markov games. Similarly, Hu *et al.* [170], Lauer *et al.* [171] and Arslan *et al.* [172] applied MARL in the game learning. Jaderberg *et al.* [140] developed a tournament-style evaluation in 3D multiplayer games, while Bard *et al.* [173] applied MARL in Hanabi as a new benchmark. Foerster *et al.* [174] presented the Bayesian action decoder(BAD) as a new public belief MDP. Lee *et al.* [175] proposed a policy evaluation with a linear approximation and actor-critic to improve the performance of MARL. Many efforts then concentrated on deep neural networks as a functional approximator in MARL [176–181]. The relative over-generalisation problem was tackled through developing a Multi-agent Soft Q-learning in continuous action spaces by Wei *et al.* [182, 183]. In addition, other works like CommNet [184], ATOC [185] and SchedNet [186] focused on exploiting an inter-agent communication. Son *et al.* [187] proposed QTRAN to acquire a more general factorisation and thus increasing the application range for MARL. Wai *et al.* [188] applied a double averaging scheme to optimise the performance of MARL. Qu *et al.* [189] introduced a value-propagation method based on a primal-dual decentralised optimisation strategy in MARL. Liao *et al.* [190] applied MARL in a 3D medical image segmentation problem. However, these aforementioned works focused on the cooperation of multi-agent systems, which strictly relied on an open environment. In addition, the multi-agent reinforcement learning just interacts with the internal agents of single RL algorithm, which cannot interact with agents of other algorithms.

The previous MARL and DQN have been applied successfully in various applications. However, the combination method of the proposed algorithm (namely RVL) is different from them, which involves the virtual part, basic part and cooperation part. Specifically, both virtual part and basic part can be applied with much flexibility. For example, the virtual part can exploit a traditional neural network and other models like practical swarm optimisation (PSO) control method, fuzzy control approach, TD model, Sara learning, Q-learning, DQN, and MARL; imitation learning and deep recurrent Q-learning algorithms can be used in the basic part. The proposed RVL is general but very effective, which can be creatively used in a wide range of methods. To show the advantages of RVL, the virtual part and the basic

part will be applied with both popular and simple prediction models and improved Q-learning method [134]. When combined with RVL, the new model consistently outperforms the original model.

4.3 Methodology

In this paper, the proposed RVL is expected to control a fed-batch process, while the main control task is to maximise the final quality. Specifically, the number of the desirable productions can be denoted as $C_t = [c_1, c_2, ..., c_t]$ by a sequence of control signals $u_t = [u_1, u_2, ..., u_t]$. For RVL, the virtual space equals to the virtual environment, which can directly replace the real environment to interact with the agent of control algorithms as the basic part. Let I_e be the virtual space of the virtual part, B be the basic function of the basic part, and RV be the final algorithm part. I_e, B and RV can be described in RVL as:

$$L^f(RV) = L^v(B_v \mid I_e) \circ L^r(B_r \mid R_e), \tag{4.1}$$

where $L^f(RV)$ denotes the optimised final learning policy, which can be acquired by a virtual learning policy $L^v(B_v \mid I_e)$ and a real learning policy $L^r(B_r \mid R_e)$; $\circ$ represents the element-wise product; R_e is the real environment space. Therefore, a virtual space I_e can create a virtual environment of fed-batch process in the virtual part. The basic functions B_v and B_r can interact with both virtual and real environments to get a virtual learning policy $L^v(B_v \mid I_e)$ and a real learning policy $L^r(B_r \mid R_e)$, and further achieve the cooperation with each other to obtain a final learning policy $L^f(RV)$. A better control signal u is also given to control the fed-batch process: $L^f(RV) \rightarrow u_t = [u_1, u_2, ..., u_t] \rightarrow C_t = [c_1, c_2, ..., c_t]$.

4.3.1 Virtual Space

An important element of RVL is the virtual space, which can create a virtual environment to interact with the agent of the basic part. With the development of the prediction models, several advanced algorithms were proposed, *e.g.*, RNN, which

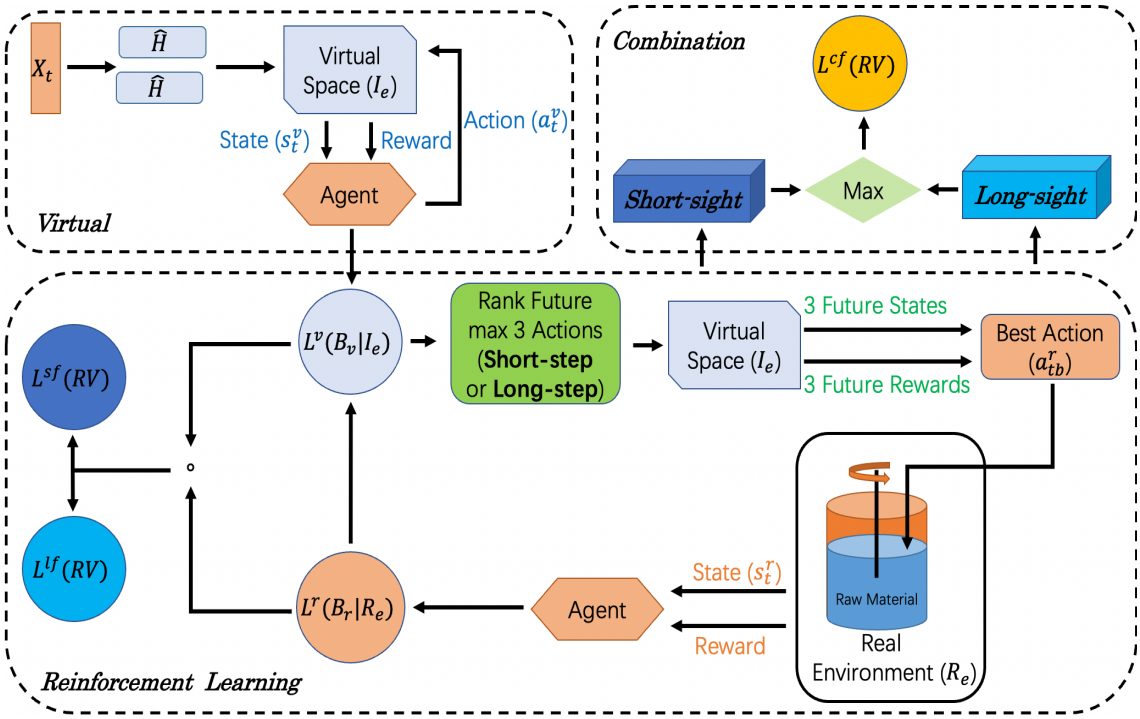


Figure 4.2: The detailed structure of our Reinforcement Virtual Learning (RVL): The virtual learning policy can be acquired by the virtual part, which interacts with a real agent by different steps to obtain the different real learning policies. After that, they are combined to get the final learning policy.

is still the most popular prediction model so far. Plenty of improved models have then been proposed based on RNN, such as Elman Network, Jordan Network, Bi-directional Long Short-term Memory Network (BiLSTM), Gated Recurrent Unit (GRU), and Long Short-term Memory Network (LSTM) [191]. Compared with the traditional RNN, these approaches have some modifications, involving gates, memory cells, and hidden states for LSTM. Specifically, based on these developments, when LSTM resolves the time-series data, it shows a better performance compared with the traditional RNN.

For fed-batch process, the short-term reaction time affects future long-term reaction. Considering a fact that both short-term and long-term information are important, and as an advanced algorithm in RNN, LSTM is resonable to model the fed-batch processes. The gates of LSTM, as the most important component to capture valid information and store them into the memory cell, the prediction of method may benefit with higher accuracy under LSTM.

We model a virtual space I_e by an LSTM model $\hat{H}$ with the historical data X_t:

$$I_e(S) = \hat{H} \mid X_t(A) = \sigma(W_y h_t + b_y) \mid X_t(A), \tag{4.2}$$

where σ is the sigmoid activation function, h_t is the hidden state, W_y and b_y are the weight and bias, A is the action space, and S denotes the state space. The current action and the next state can be indicated by a_t and s_{t+1}, respectively,

$$a_t \, (\, a_t \in A) \rightarrow I_e(S, A) \rightarrow s_{t+1} \, (\, s_{t+1} \in S), \tag{4.3}$$

where the next state s_{t+1} can be obtained through I_e model by the selected current action a_t.

4.3.2 Reinforcement Virtual Learning (RVL)

Virtual Leaning Policy

This part provides the strategy of the interactions between the virtual space and RL agent. In terms of the modelled virtual space I_e, the agent of RL can generate the virtual state after interaction with I_e. Then, a virtual learning policy $L^v(B_v \mid I_e)$ can be acquired through a virtual basic function B_v:

$$B_v = E\left\{\hat{E}_t \mid I_e(s_t^v, a_t^v) \, , s_t^v \in S \, , a_t^v \in A)\right\}. \tag{4.4}$$

Here, s_t^v and a_t^v denote the virtual state and the action, respectively. $\hat{E}_t$ represents the expected reward:

$$\hat{E}_t = \sum_{n}^{\infty} \gamma^n r_{t+n}, \tag{4.5}$$

where the expected gains are denoted by r_{t+n} and γ $(0 < \gamma < 1)$ is the discount factor. Following Eq. (4) and Eq. (5), the virtual learning policy $L^v(B_v \mid I_e)$ can be described as

$$\begin{aligned} L^v(B_v \mid I_e) \leftarrow B_v(s_t^v, a_t^{m_k v}) + \alpha[r_{t+1}^v + \\ \gamma^v \max_{a_t^{m_k v} \in A} B_v(s_{t+1}^v, a_t^{m_k v}) - B_v(s_t^v, a_t^{m_k v})], \end{aligned} \tag{4.6}$$

where α $(0 < \alpha < 1)$ indicates the learning rate, and $a_t^{m_k v}$ describes that a virtual action a_t^v can be executed m time steps in k^{th} period based on SMSA [134]. r_{t+1}^v denotes the virtual expected benefits. The maximum virtual value at next virtual state s_{t+1}^v is then represented by $\max_{a_t^{m_k v} \in A} B_v(s_{t+1}^v, a_t^{m_k v})$. Considering that the agent can interact with different environments, the weight of benefits is therefore distinguishable for RVL in different environments. Following this principle, we set different discount factors, where γ^v represents the virtual discount factor in a virtual environment.

Real Leaning Policy

It is worth noting that the agent can acquire a virtual learning policy with a virtual environment, which means RL can be learnt in an unknown and uncertain environment. Based on this, the learned virtual learning policy can further guide the agent to learn a real learning policy $L^r(B_r \mid R_e)$, when the agent interacts with a real space R_e. Specifically, interacting with a real environment, the current best real action a_{tb}^r at the current real state s_t^r can be predicted based on the results of future steps by a virtual learning policy $L^v(B_v \mid I_e)$ combined with a virtual environment.

For instance, three actions a_{t1}^v, a_{t2}^v, a_{t3}^v at the current state s_t^v can be obtained based on the virtual learning values $B_v(s_t^v, a_{t1}^v), B_v(s_t^v, a_{t2}^v), B_v(s_t^v, a_{t3}^v)$ of a virtual learning policy $L^v(B_v \mid I_e)$ in terms of maximum to minimum:

$$\begin{aligned} B_v(s_t^v, a_{t1}^v), B_v(s_t^v, a_{t2}^v), B_v(s_t^v, a_{t3}^v) \leftarrow \\ \max_{A_t^v \in A} B_v(s_t^v, A_t^v) \mid L^v(B_v \mid I_e), \end{aligned} \tag{4.7}$$

where A_t^v indicates all possible actions at state s_t^v. Based on the virtual learning policy, the agent can know several suitable actions in each state. In this paper, three suitable actions are enough for the task. However, the agent cannot immediately determine the best action from them. The agent needs to select three actions to interact with the virtual environment to reach three different next-states $s_{t1}^v, s_{t2}^v, s_{t3}^v$, respectively. After that, the agent can follow $L^v(B_v \mid I_e)$ to reach three different future states of N steps $s_{N1}^v, s_{N2}^v, s_{N3}^v$. Different future-states can show the perfor-

mance of control by the proposed algorithm, which can then be reflected by the expected benefit (reward) of each different state:

$$r_N^v(s_N^v) = \max(r_t^v + (r_N^v(s_{N1}^v, a_{t1}^v), r_N^v(s_{N2}^v, a_{t2}^v), r_N^v(s_{N3}^v, a_{t3}^v)) \mid L^v(B_v \mid I_e), \tag{4.8}$$

where $r_N^v(s_N^v)$ is the maximum reward obtained after N future steps. If the maximum reward is $r_N^v(s_{N1}^v, a_{t1}^v)$, the best state is s_{N1}^v, which means the best action a_{tb}^r is a_{t1}^v at state s_t^v and s_t^r. Following this principle and the basic function B_r in real space R_e, the virtual learning policy is similar to that of $L^r(B_r \mid R_e)$ to learn:

$$\begin{aligned} L^r(B_r \mid R_e) \leftarrow B_r(s_t^r, a_{tb}^{m_k r}) + \alpha[r_{t+1}^r + \\ \gamma^r L^v(B_v \mid I_e)(s_{t+1}^r, a_{tb}^{m_k r}) - B_r(s_t^r, a_{tb}^{m_k r})], \end{aligned} \tag{4.9}$$

where the real expected benefits is denoted as r_{t+1}^r. The real states and actions are represented by s_t^r and a_t^r. In addition, the real discount factor is represented as γ^r. In the real space, the virtual learning policy $L^v(B_v \mid I_e)(s_{t+1}^r, a_{tb}^{m_k r})$ guides the agent to learn a real learning policy $L^r(B_r \mid R_e)$, which gives both feedback and the cooperation about the previous $L^v(B_v \mid I_e)$ to obtain a new final virtual learning policy $L^v(B_v \mid I_e)$:

$$\begin{aligned} L^v(B_v \mid I_e) \leftarrow B_v(s_t^v, a_t^{m_k v}) + \alpha[r_{t+1}^v + \\ \gamma^v L^r(B_r \mid R_e)(s_{t+1}^v, a_t^{m_k v}) - B_v(s_t^v, a_t^{m_k v})]. \end{aligned} \tag{4.10}$$

Finally, the optimised final learning policy function $L^f(RV)$ can be acquired by the cooperation between the learned virtual learning policy $L^v(B_v \mid I_e)$ and the real learning policy $L^r(B_r \mid R_e)$.

Combination

When the agent selects the best action at each state in a real environment, the future states of some steps can further influence the selection of the actions based on a virtual learning policy $L^v(B_v \mid I_e)$. Therefore, the future step is important for the proposed RVL, as well as the combination. For the future step, the states of

the first step and the final step influence the choice of the best action at the current state. Here, the first step (named as 1-step) is referred to as the short-sight and the final step is denoted as the long-sight.

The framework is summarised in Figure 4.2. Specifically, through both the short-sight and the long-sight steps, a final short-sight learning policy $L^{sf}(RV)$ and a long-sight learning policy $L^{lf}(RV)$ can be obtained. After that, the maximum combination can be executed to acquire the final combination learning policy $L^{cf}(RV)$:

$$L^{cf}(RV)(s_t^{cf}, a_t^{cf}) = \max(L^{sf}(RV)(s_t^{sf}, a_t^{sf}), L^{lf}(RV)(s_t^{lf}, a_t^{lf})). \tag{4.11}$$

For RVL, the virtual space can interact with a basic function online to obtain the virtual knowledge. The learned information can further guide the real agents to learn real knowledge, when the agents interact with a real environment. The agents acquire useful knowledge through the virtual knowledge, thereby improving the efficiency of exploration in a real environment. The feedback of the real knowledge modifies the virtual knowledge, such that more accurate virtual knowledge can help the real agents to acquire better real knowledge. In this work, the real knowledge can be obtained effectively, resulting in the better performances for the original algorithms. Furthermore, RL can be applied directly without certain and known environments as the proposed virtual space.

4.4 Experiments

We design different sets of experiments to verify the performance of the proposed method. The advantages of the new algorithm can be shown directly by our control results, where the key results are analysed below. In the experiment, we applied the simulation data to replace real data. The reason is that existing simulation function have been reflected the real reaction progress. The proposed method can only be indirectly validated for real-world applications once it demonstrates strong performance in simulated data. Meanwhile, using simulation data has cost efficiency.

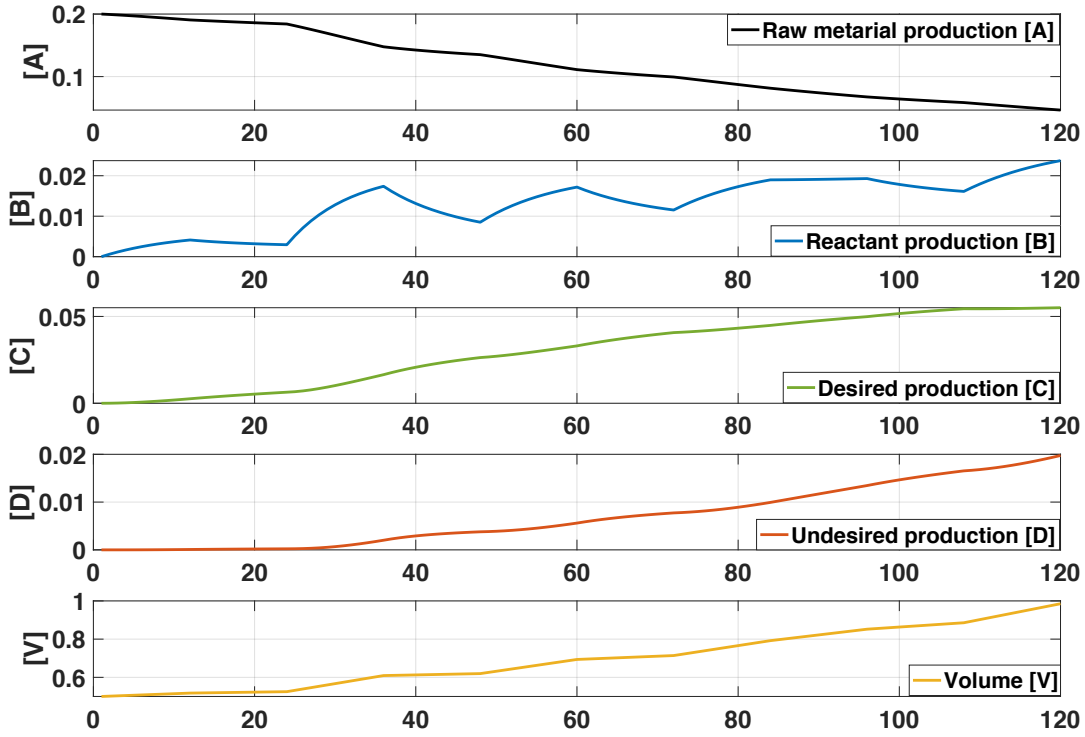


Figure 4.3: An example sequence of [A], [B], [C], [D], [V] during a reaction process based on real environment

4.4.1 Set-up for the Dataset

Fed-batch Process Model

Although a number of control algorithms were applied in chemical processes, machine learning-based methods were explored barely in recent years. It is worth noting that machine learning-based control results are often superior to others, which means machine learning-based technique can be applied successfully in chemical processes. As a traditional process of chemical processes, the batch process is important. The main strategy is that the proposed algorithm can control it optimally as shown in our experiments.

The fed-batch process is a classical batch process, and therefore, we apply it in this work. This fed-batch process is described as follow:

$$A + B \xrightarrow{k_1} C, \tag{4.12}$$

$$B + B \xrightarrow{k_2} D, \tag{4.13}$$

where the reactants A and B are the raw materials; C and D are the desirable productions and the undesirable by-products, respectively. The reactant B can be added into the reactor gradually, to prevent the fast formation of the undesirable by-products D during the specified batch time $t_f = 120\ min$.

In the fed-batch process, the main control purpose is that the desirable products C should be acquired as much as possible, while the undesirable products D should be kept at the lowest quantity in the whole reaction batch time, where the total volumes V cannot exceed 1 m^3.

In the control task, the concentration of reactant B is added in a feed stream with concentration $b_{feed} = 0.2$. The following fed-batch process model is developed based on the material balances and the reaction kinetics:

$$\begin{aligned}
\frac{d[A]}{dt} &= -k_1[A][B] - \frac{[A]}{V}u, \\
\frac{d[B]}{dt} &= -k_1[A][B] - 2k_2[B]^2 + \frac{b_{feed} - [B]}{V}u, \\
\frac{d[C]}{dt} &= -k_1[A][B] - \frac{[C]}{V}u, \\
\frac{d[D]}{dt} &= 2k_2[B]^2 - \frac{[D]}{V}u, \\
\frac{d[V]}{dt} &= u.
\end{aligned} \tag{4.14}$$

The concentrations of A, B, C and D are represented by $[A]$, $[B]$, $[C]$ and $[D]$, respectively. The volume of the materials in the reactor and the reactant feed rate are denoted by V and u, respectively. The reaction rates are represented as k_1 and k_2, and are set to 0.5, as shown in Table I. The initial $[A]$ is 0.2 moles/litter and $[V]$ is 0.5. Based on the above model, a simulation program of the fed-batch process can be developed using Matlab, and the simulation is used to test the various control algorithms. In this paper, the simulation of fed-batch process is called the real reaction process.

In terms of the real reaction process and an example sequence $[A]$, $[B]$, $[C]$, $[D]$, $[V]$ is shown in Figure 4.3.

Dataset

The dataset is constructed by 20,000 sequences, relying on the base fed-batch process model in our experiments.

Let sequences control signals be $U_t^i = [u_t^1, u_t^2, u_t^3, ..., u_t^i]$, the desired productions be $C_t^i = [c_t^1, c_t^2, c_t^3, ..., c_t^i]$, the undesired productions be $D_t^i = [d_t^1, d_t^2, d_t^3, ..., d_t^i]$, and the constructed historical information X_t. We randomly select 15,000 sequences data as our training data, and the remaining 5,000 sequences are taken the test data.

For desired product C, the prediction model $\hat{H}$ has 100 hidden neurons in hidden state layer and the mini-batch size is set to be 20, then this model is trained by 3,000 epochs. Compared with desired product C model, the prediction model of undesired product D has 200 hidden neurons and the training time is 6,000 epochs.

4.4.2 Reinforcement Virtual Learning Design for Fed-batch Process

In this paper, RVL is based on the traditional RL, such that the important construction elements of RVL are similar to the traditional Q-learning. Therefore, the models of agent, state, action and reward function are vital as well.

The Agent Design

As an element of RVL, several important parameter of the RL model (e.g., learning rate α and discount factor γ) should be set first. For the proposed algorithm, two different learning policies will continuously interact during the learning time with two different discount factors γ^v and γ^r. Specifically, the virtual learning policy $L^v(B_v \mid I_e)$ of the virtual space I_e is trained by the prediction model $\hat{H}$, and its discount factor γ^v influences less than that of in the real learning policy $L^r(B_r \mid R_e)$ of the real environment R_e after $L^v(B_v \mid I_e)$. The discount factor γ^r is expected to significantly affect the final learning policy $L^f(RV)$. In addition, as an essential part of RVL, the ϵ-*greedy* policy needs to be set with a suitable ϵ value. Table 4.1 denotes these parameters for our experiments.

Table 4.1: Parameters used in the simulations.

Variable	***Meaning***	***Setting***
k_1, k_2	reaction rate	0.5
α	learning rate	0.1
γ^v	virtual discount factor	0.7
γ^r	real discount factor	0.98
ε	greedy-probability	0.7

Table 4.2: States of the fed-batch process

Condition	***State***
$\Delta[C] - \Delta[D] \geq 0.0008$	S_1
$0.0007 \leq \Delta[C] - \Delta[D] < 0.0008$	S_2
$0.0006 \leq \Delta[C] - \Delta[D] < 0.0007$	S_3
$0.0005 \leq \Delta[C] - \Delta[D] < 0.0006$	S_4
$0.0004 \leq \Delta[C] - \Delta[D] < 0.0005$	S_5
$0.0003 \leq \Delta[C] - \Delta[D] < 0.0004$	S_6
$0.0002 \leq \Delta[C] - \Delta[D] < 0.0003$	S_7
$0.0001 \leq \Delta[C] - \Delta[D] < 0.0002$	S_8
$0 \leq \Delta[C] - \Delta[D] < 0.0001$	S_9
$\Delta[C] - \Delta[D] < 0$	S_{10}

The State Design

In our experiments, the main control purpose is that the desirable products $[C]$ are produced as much as possible, while the undesirable by-products $[D]$ should be kept at a low quantity at the end [134]. We design the state based on this principle. During the given reaction time, each product goes through some fluctuations following the implementation of the control policy. Once the increasing rate of the desirable product concentration is high, more desired products are expected to be produced, while the increasing rate of the undesirable products should be kept low at the meantime. Following this principle, the slope of $[C]$ curve should be steeper

than that of $[D]$ curve, to achieve a desired result in the whole reaction time. Therefore, the state can be represented by the differences in derivatives between $[C]$ and $[D]$ as described in Table 4.2, where $\Delta[C]$ and $\Delta[D]$ are the slope of $[C]$ and $[D]$, respectively.

The Action Design

In our experiments, $[A]$ is given at the beginning of the reaction. With the adding of u, $[B]$, $[C]$, $[D]$ and $[V]$ are changed. Therefore, the feeding rate u decides the the control signal and the action space in the range of from 0.001 to 0.009.

The Expected Benefit Function Design

For any algorithms of RL, the design of the benefit is one of the most important part. Actually, the benefit and the punishment of the expected benefit function can directly influence the learning performance of the algorithm, resulting in a flexible design of the expected benefit function.

The agent can predict future results accurately by the virtual part, leading to the improvement of the the accuracy of the expected benefit and its selection of action in the whole learning process for the proposed RVL. In this case, a direct and simple benefit function is approximated, with the design of the expected benefit function represented by a constant value based on the traditional methods. In this fed-batch process, the distribution of the expected benefit function is followed by a state space.

4.4.3 The Control Results

Experimental Details

Considering a fact that RVL creates the virtual learning policy $L^v(B_v \mid I_e)$, which can predict the estimated future results to indicate and interact with the real agent to further acquire a better real learning policy $L^r(B_r \mid R_e)$. Based on this principle, RVL will predict some future steps during the control and the learning processes. We set several experiments based on the virtual 1-step, 30-step, 50-step, 80-step and

Table 4.3: The control results of RVL compared with other control algorithms

Algorithm	*[C]*	*[D]*	*[V]*	*[C]-[D]*	*([C]-[D])*[V]*
Recurrent neuro-fuzzy network [127]	0.0559	0.0304	0.9900	0.0355	0.0351
Nominal control [127]	0.0615	0.0345	0.9918	0.0267	0.0264
Minal risk [127]	0.0612	0.0236	1.000	0.0376	0.0376
Q-learning [134]	0.0590	0.0193	0.9220	0.0366	0.0366
SMSA [134]	***0.0618***	0.0236	0.9800	0.0361	0.0361
RVL	0.0614	***0.0199***	***0.9254***	***0.0415***	***0.0384***

120-step (final step). After that, the proposed combination-step experiments will be applied as well.

Comparison with Other Algorithms

To show the control performances, the control results of RVL are directly compared with other state-of-the-art control algorithms, such as the recurrent neuro-fuzzy network, traditional Q-learning, stochastic multi-step action Q-learning (SMSA) [134], nominal control, and minimal risk control algorithm [127]. Table 4.4.2 shows the results of different control algorithms.

In Table 4.4.2, although more desirable productions $[C]$ are produced by the nominal control and the SMSA algorithm when compared with other algorithms, more undesirable productions $[D]$ are generated as well. For RVL, we note that the difference between $[C]$ and $[D]$ is maximum, and the difference between desired final species $[C][V]$ and undesired final species $[D][V]$ is also maximum. This indicates that when the proposed RVL algorithm achieves the best compared with other control algorithms.

In summary, the final control algorithm will follow the combination of the short-long step based on RVL. The control results of $[C]$ and $[D]$ are described in Figure 4.4, and the final suitable control signal under the proposed algorithm is shown in Figure 4.4.

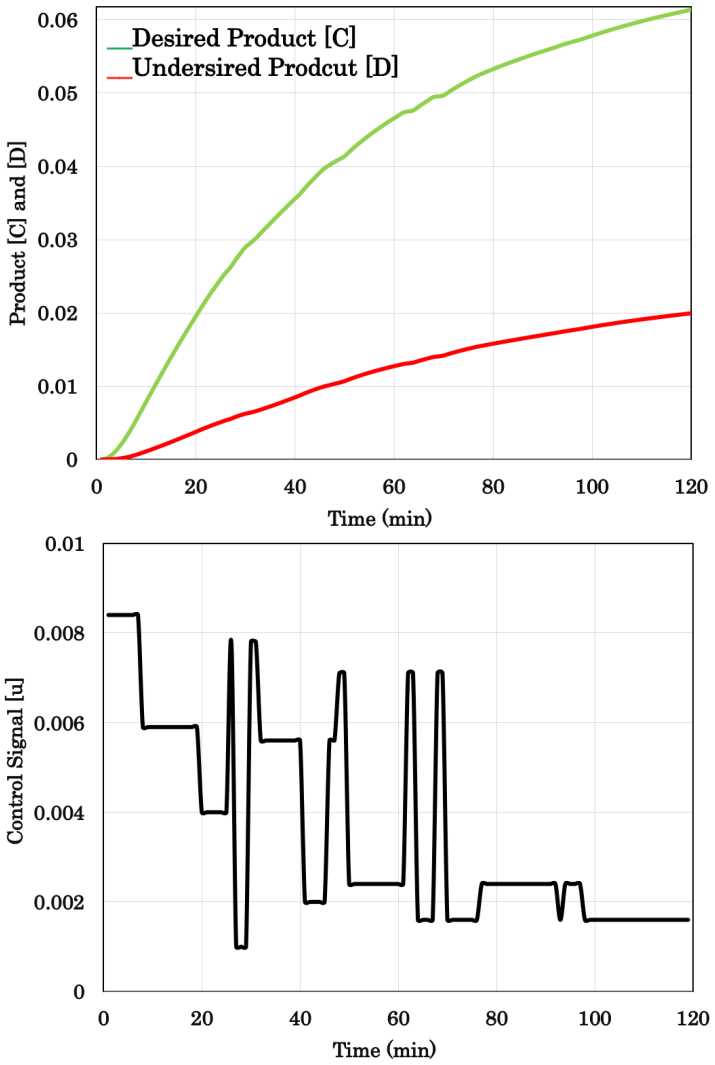


Figure 4.4: The variation during a reaction process of the desirable products [C] and the undesirable products [D] based on RVL control. The control signal [u] under RVL control.

4.4.4 Detailed Evaluations

The Virtual Prediction Results

Figure 4.7 presents the prediction and the ground truth of the desirable products $[C]$ and the undesirable products $[D]$. We observe that both products can be predicted accurately under the model $\hat{H}$.

The Root Mean Squared Error (RMSE) between the predictions and the test data of two different productions are shown in Figure 4.5 and Figure 4.6. It shows that the trained desirable product model C and the undesirable product model D under $\hat{H}$ can predict the real reaction process of $[C]$ and $[D]$ accurately. In summary,

Table 4.4: The control results of [C] and [D] based on different pure steps.

Algorithm	***[C]***	***[D]***	***[V]***	***([C]-[D])*[V]***
1-step	0.0606	0.0182	0.8999	***0.0381***
30-step	0.0558	0.0173	0.9433	0.0363
50-step	0.0566	0.0179	0.9638	0.0372
80-step	0.0579	0.0218	1.0000	0.0361
120-step	0.0613	0.0211	0.9254	***0.0372***

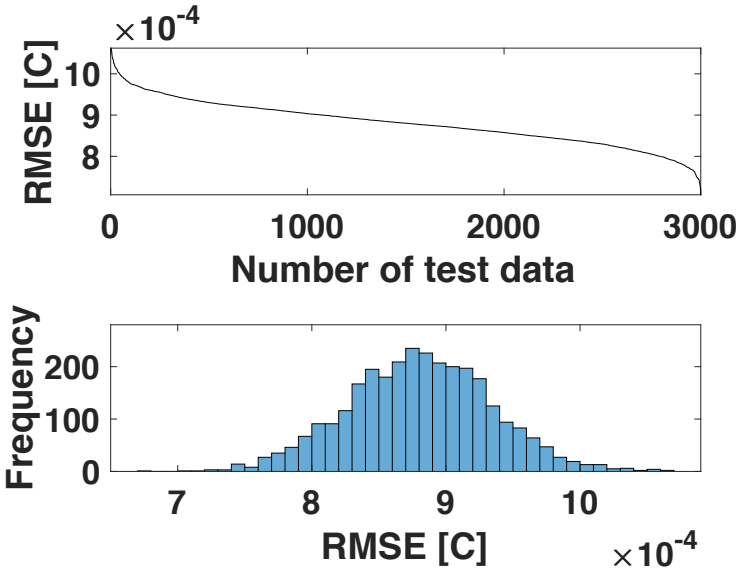


Figure 4.5: RMSE between the predictions and the ground truth of the desirable products [C].

the model $\hat{H}$ shows the outstanding performance, which can clearly predict the variations of the desirable products $[C]$ and the undesirable products $[D]$ under different control signals for the fed-batch process. Therefore, the trained models for C and D can be referred as the virtual reaction process, which can replace the real reaction process, especially for learning the virtual learning policy.

Impact of Step Size

In order to describe the control performance of the proposed RVL algorithm, the results of different pure steps are shown: short-sight (1-step), immediate-sight (30-

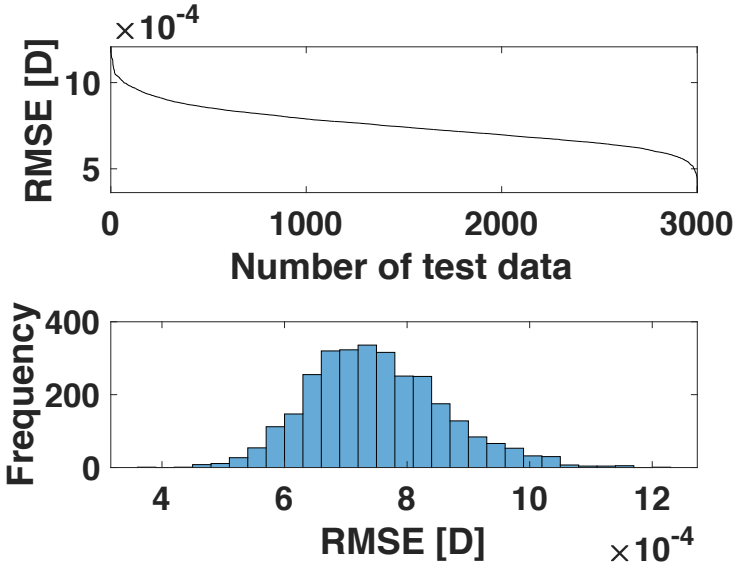


Figure 4.6: RMSE between the predictions and the ground truth of the the undesirable products [D].

Table 4.5: The control results based on different combination-steps

Algorithm	***[C]***	***[D]***	***[V]***	***([C]-[D])*[V]***
Short-Immediate step	0.0603	0.0173	0.8898	0.0382
Immediate-Long step	0.0601	0.0171	0.8913	0.0383
Short-Long step	***0.0614***	0.0199	0.9254	***0.0384***

step, 50-step and 80-step) and long-sight (120-step), respectively. The control results of the combination steps of different sights are reported as follows. Table 4.4 indicates the results of the desirable products $[C]$ and the undesirable products $[D]$ based on different pure steps. Figure 4.8 describes the variation curves.

In Figure 4.8 and Table 4.4, we can observe that when the agent predicts 1-step and 120-step, more desirable productions $[C]$ can be acquired compared with other steps of control, which means the short-sight and long-sight have a better control performance.

Secondly, the combination-step will be applied. In our experiments, the learning policy of short-sight will be combined with that of the immediate-short and the long-

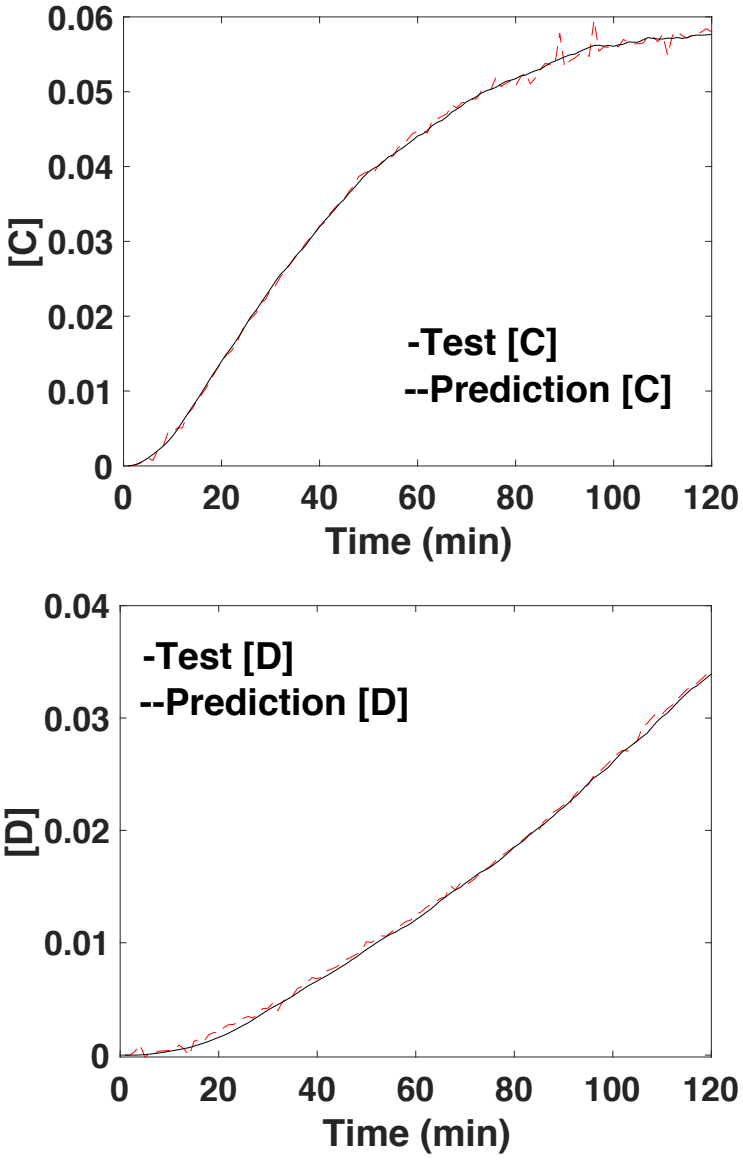


Figure 4.7: The prediction and the ground truth of the desirable products [C] and the undesirable products [D].

Table 4.6: The total expected benefits of different steps of algorithms.

Algorithm	***Total Expected Benefits***
1-step (Short-sight)	33100
30-step (Immediate-sight)	6500
50-step (Immediate-sight)	16200
80-step (Immediate-sight)	7900
120-step (Long-sight)	26700
Short-Immediate Combination	***34400***
Immediate-Long Combination	***33400***
Short-Long Combination	***30800***

sight, respectively. When the algorithm applies the combination-step, we acquire a better performance as shown in Figure 4.9 and Table 4.4.4.

Specifically, there are more desirable productions $[C]$ and less undesirable productions $[D]$ after being applied the combination-step compared with the pure immediate-step (30-step, 50-step, 80-step) and the long-sight step (120-step).

Once the combination-step is applied, the improvement for control can be proved by the total expected benefits. When the combination-step is applied, the total reward will be increased compared with different pure steps. We demonstrate the details in Table 4.4.4.

Following Table 4.4.4, 1-step (short-sight) and 120-step (long-sight) can collect more expected benefits than immediate-sight for the pure step, which indicates that the control results of short-sight and long-sight are better as shown in Figure 4.9 and Table 4.4.4. In addition, it also proves that the expected benefits can reflect the performance of RL and control results. Obviously, the combination-step can acquire more expected benefits in total compared with different pure steps, and thus, the performance of learning policy and control of combination-step will be better. Especially, the combination-steps of short-immediate and immediate-long-sight can be improved significantly compared with immediate-sight (30-step, 50-step and 80-step).

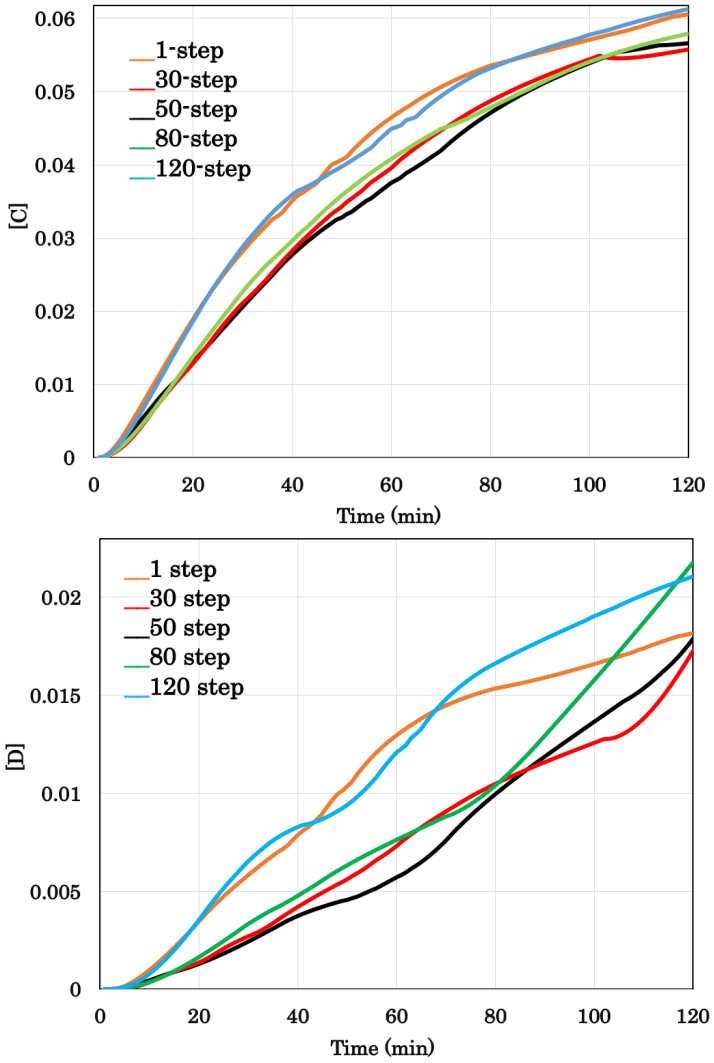


Figure 4.8: The variation curves of [C] under different pure steps control and [D].

Although the total expected benefits of 1-step, short-immediate, and immediate-long combination-step are greater than the short-long combination-step, the final control result of short-long combination-step is the best. The reason is that the expected benefits can be acquired easily in the previous and the immediate reaction time (short-sight and immediate-sight) compared with the latter reaction time (long-sight) in terms of the state. The expected benefit function is shown in Figure 4.10.

In this fed-batch process, the differences between $\Delta[C]$ and $\Delta[D]$ (the value of the state) during the previous and the immediate reaction time are greater than that in the latter reaction time. Therefore, more expected benefits can be obtained by the 1-step, short-immediate and immediate-long combination-step compared with the short-long combination-step. However, the control policy of the short-sight and the

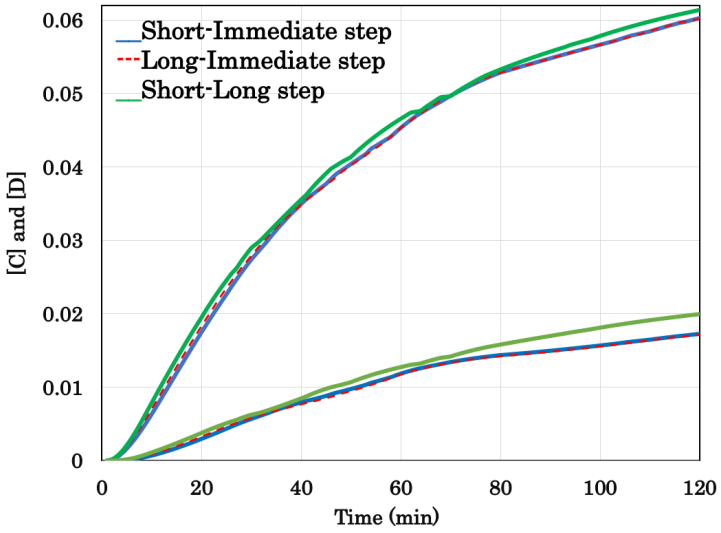


Figure 4.9: The variation curves of the desirable [C] and undesirable [D] under different combination steps of control.

immediate-sight can emphasise the short and the immediate control results, resulting in a better performance, while the final control results are not the best ones. On the contrary, the long-sight can pay more attention to the final results, and thus generating better final control results. When long-sight is combined with short-sight, the control policy can emphasise both previous and latter control results, and therefore, the control performance of the short-long combination-step is the best. Based on the comparisons with other algorithms, the proposed RVL can achieve the best control results.

4.5 Conclusion

In this chapter, we proposed a novel Reinforcement Virtual Learning (RVL) algorithm by creating a virtual space to interact with the agent of RL and the learned virtual policy. The agent of RL can be introduced to learn the real learning policy resulting the feedback to modify the virtual learning policy after interaction with real environment. It is worth noting that the approximated future results of the combinations between short-sight and long-sight through the virtual environment can help the agent to acquire a better real control policy. The proposed RVL overcomes several existing problems, such as uncertain environment, time-variation, and

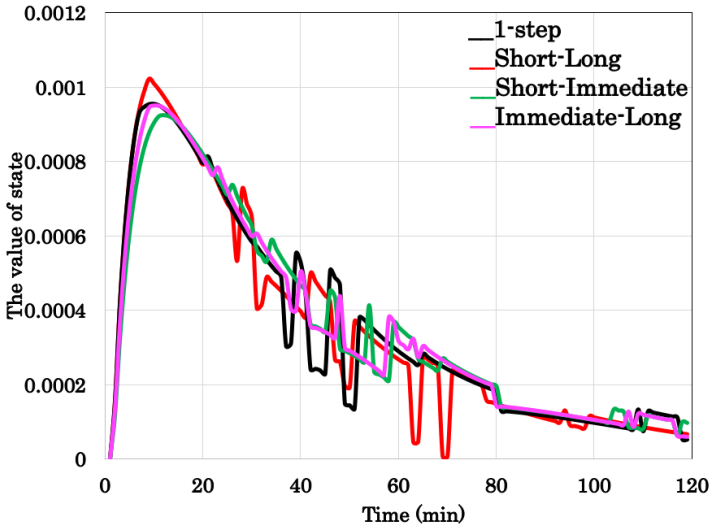


Figure 4.10: The the expected rewards at different steps during the whole reaction time.

non-linearity. In addition, our experiments demonstrated that the fed-batch process controlled by the proposed RVL can outperform the existing stare-of-the-art algorithms, leading to the effective and stable control performances.

Further work includes applying the proposed RVL to other control applications. For example, RVL can be served for robot control by learning a virtual strategy through a virtual environment of RVL. In addition, this virtual strategy can help the robot to achieve some control tasks. Inverse reinforcement learning can replace the basic part of RVL, which can be applied in self-driving as well. When MARL and CNN are applied in both the virtual part and the basic part, they can tackle some medical issues. The proposed RVL can be combined with graph neural networks in some applications as well.

CHAPTER 5

Sequential Visual Information in Video Inpainting

Beyond sequential process control, computer vision techniques also involve significant sequential information. There are several motivations for moving from reinforcement learning in chemical batch process control to video inpainting. One motivation is to testify deep learning techniques in a new and different domain and see study the flexibility of the system deployment. This could provide an opportunity to explore new challenges and develop new approaches to solving problems. Another motivation is that video inpainting is an important problem in computer vision and has applications in areas such as security, entertainment, and healthcare. In process control, sequential information has been used to predict future results. By applying a similar idea to video inpainting, we believe the visual contexts surrounding the inpainting area can also be modelled as a sequential relationship in the spatial domain. Additionally, there are technical and methodological similarities between reinforcement learning in chemical batch process control and video inpainting. For example, both problems may involve optimising a sequence of actions to achieve a desired outcome, leveraging prior information, and dealing with uncertainty.

There are differences between the two areas also lead to technical challenges. One key difference is the nature of the problem. Batch process control involves

controlling a physical system to produce a desired output, while video inpainting involves filling in missing or corrupted parts of a video sequence. This means that the types of algorithms and techniques used in each field may differ. Another difference is the level of uncertainty involved in the problem. In batch process control, there may be uncertainty in the dynamics of the system, such as fluctuations in temperature or pressure, which must be accounted for in the control strategy. In video inpainting, the uncertainty may come from incomplete or noisy data, such as missing or corrupted frames. The types of data and measurements involved also differ between the two fields. In batch process control, the data may come from sensors that measure physical parameters such as temperature, pressure, and flow rates. In video inpainting, the data may come from video cameras or other imaging sensors.

Despite these differences, both batch process control and video inpainting require intelligent decision-making over time to achieve the desired outcome. Both fields may benefit from the use of sequential decision-making techniques, such as reinforcement learning, to optimise performance and achieve better outcomes. In video inpainting, machine learning can be used to learn the patterns and structures of the surrounding pixels and fill in missing or corrupted regions. Both fields rely on data to make decisions. Data from previous frames or other sources can be used to fill in missing or corrupted parts of the video sequence. Both batch process control and video inpainting may involve highly nonlinear optimisation problems. In video inpainting, the highly nonlinear relationships between the surrounding pixels and the missing or corrupted regions can make it challenging to fill in the missing data accurately.

Overall, while batch process control and video inpainting are different from intuition, they share many common technical aspects that require intelligent decision-making, optimisation, machine learning, data-driven approaches, and nonlinear optimisation techniques. This chapter will thoroughly observe how the two distinctive research domains can be unified in the sequential information framework.

5.1 Introduction

Video inpainting aims to complete blank regions of each video sequence frame with plausible and coherent content. It is widely adapted to real-world applications, such as video restoration [192], logo removal [193], video editing [194], and video stabilization [195]. Specific, the Figure 5.1 shows the main task of video inpainting. The first row displays three normal frames of a video sequence, while the third row represents the masked frames. The main task of video inpainting is that model needs generate background pixels of each frames filling masked football and boy to obtain a high quality video sequence. Despite the outstanding progress made in image inpainting [48, 49, 63], video inpainting remains challenging due to the complicated object motion and dynamic camera motion in video frames. Directly applying image inpainting approaches on each video frame tends to generate inconsistent videos, since the temporal reliance information has been largely neglected. In this case, the idea of considering both contents and temporal coherence for synthesizing high-quality video frames motivates researchers to exploit more effective approaches for the video inpainting task.

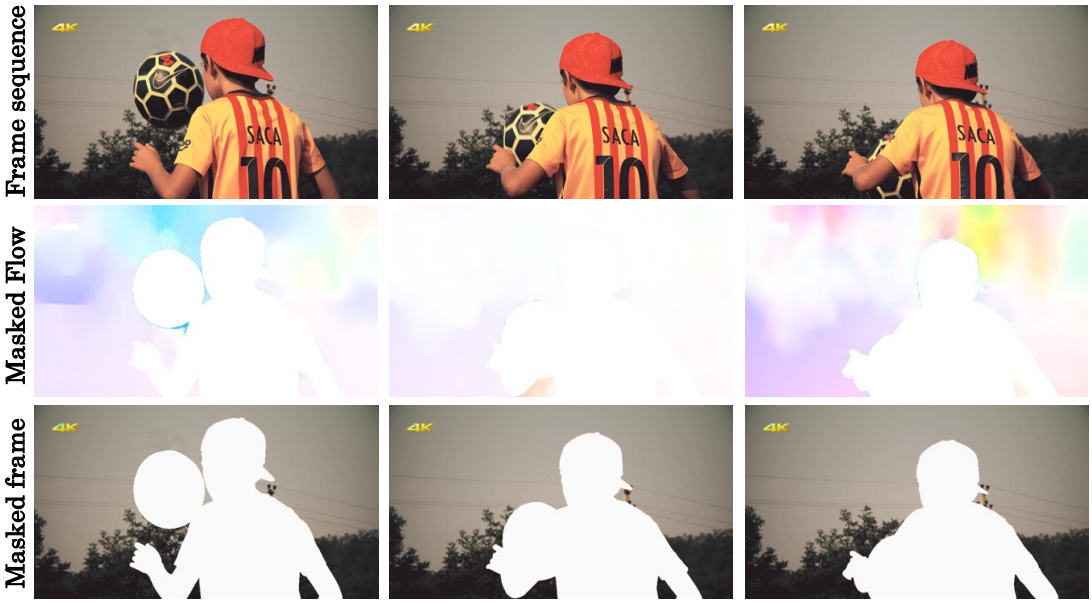


Figure 5.1: Example of input of our framework. The proposed FDTN framework aims to take the masked optical flow (second row shown in the figure) and the masked frame sequences (third row) as input and output the original frame sequence (first row).

Recently, plenty of video inpainting approaches [50,196,197] have been proposed to encode the temporal information for video synthesizing by feeding a large amount of RGB frames into 3D Convolutional Neural Networks (3D-CNN). However, they suffer from temporal artifacts due to limited temporal receptive fields [193]. To overcome this problem, many efforts used attention module obtain long-range correspondences. To preserve the temporal coherence, a few optical-flow guided approaches [51,63] have been proposed by integrating the temporal motion information from optical-flow sequences with only one single reference video frame to synthesize corrupted area. Nevertheless, those optical-flow guided approaches unintentionally neglect the content/spatial information from corrupted video frame sequences, which results in coarse synthesized video frames.

Regarding the previous flaws, in this paper, we carefully design a trainable Flow enhanced Dual spatial-temporal Transformer (FDTN) for the end-to-end video inpainting task. The proposed FDTN approach integrates an attention-wise fusion mechanism for spatial-temporal information cross-complementation, resulting in more comprehensive image synthesizing.

In summary, the main contributions of the FDTN algorithm are summarised:

- We first introduce a dual transformer-based framework for video inpainting which takes multiple modalities (pixel image and optical flow) to enhance the spatial-temporal knowledge exploration.

- We present a novel attention-wise fusion module to perform the information complementation from the optical flow modality to the pixel image modality and vice versa. The attention-wise fusion module encourages the information utilization between content and spatial-temporal information, which results in more comprehensive image synthesizing.

- Extensive experiments demonstrate the superiority of FDTN over state-of-the-art approaches in video inpainting, both qualitatively and quantitatively.

5.2 Related Work

In order to develop high-quality video inpainting algorithms, many efforts have been made to fill missing regions with spatial and temporal content in videos. We discuss representative direct-based methods, learning-based methods, and optical flow-based models for video inpainting as follows.

Direct-based methods. Traditional key methods can obtain direct appearance knowledge from known regions to complete the image blanking. Following this technology, some algorithms complete targeted missing blanks of the image by applying patches from related known regions and other relevant images [198–201]. Compared with image inpainting, the main core challenge of the video inpainting task lies in the temporal domain. Therefore, some works that execute the traditional aforementioned algorithms to solve the video inpainting task are not enough [49, 202]. To solve the difficulty of dynamic videos, the motion field was applied in the blank regions [48, 203]. However, the sophisticated and high computational is the bigger challenge and limitation. In addition, high-level semantic knowledge cannot be acquired through direct-based algorithms.

Learning-based methods. In recent years, many efforts have applied learning-based algorithms for video inpainting to overcome these limitations. Firstly, some works directly executed neural networks to apply in inpainting [204, 205]. VInet [56] applied recurrent networks for ensuring temporal coherence. Beyond applying naive neural networks, several efforts focused on the CNN algorithms to generate visuals such as Generative Adversarial Networks (GAN). Based on GAN, the large-scale missing blank regions can be completed by a trained inpainting network by Pathak [206]. Meanwhile, the proposed LGTSM [59] has a temporal shift module and spatial temporal adversarial loss to overcome spatial and temporal coherence. Recently, Iizuk and Yu improved the GAN through adversarial losses of global and local discriminators and attention algorithms in inpainting tasks [207, 208]. Lee et al. used frame-wise attention based on weighted summing of each frame [57], but it was hard to model the complex motions that solely relied on affine transformations of global frames. The missing regions could be filled following pixel-wise attention step by step [209]; however, the consistent attention result of each recursion was hard

to ensure. In terms of previous efforts, Zeng and Fu et al. applied self-attention algorithms into the GAN structure to add the temporal contents [193].

Optical-flow based methods. However, these methods cannot fully describe temporal information. Hence, many works applied optical-flow, which is the pattern of apparent motion of object, to complete the missing areas of the images in the given video [48, 210, 211]. Generally, most works applied the FlowNet2.0 [212] to extract the optical-flow. It applid CNNs to build FlowNetC, FlowNetS and FlowNet-SD with warping layers to specializing on small motions. The optical-flow was predicted with the mask to propagate the pixels of blank regions [51] and [63] following Xu's work to improve the edge details performance of a video followed by motion edges network and gradient-domain process [63]. Although these algorithms obtained good performance, it is rare that exciting video inpainting algorithms cover both enough spatial and fully temporal content.

5.3 Methodology

Given a corrupted video sequence $\mathcal{X}^T = \{\boldsymbol{x}_1, ..., \boldsymbol{x}_T | \boldsymbol{x}_T \in \mathcal{R}^{H \times W \times 3}\}$ with sequence length T and corresponding frame-wise binary masks $\boldsymbol{M}^T = \{\boldsymbol{m}_1, ..., \boldsymbol{m}_T | \boldsymbol{m}_T \in \mathcal{R}^{H \times W \times 1}\}$, we aim at synthesizing the faithful content within the corrupted (masked) areas under the optical-flow modality information enhancement.

In the following, we discuss the main components of our method. First, we utilize a flow extractor $\mathcal{F}_e(.)$, which encodes all corrupted frames into the masked optical-flow sequence, i.e., $\mathcal{X}^F = \mathcal{F}_e(\mathcal{X}^T)$, for temporal information provision at subsequent processing. Second, we extract the representations from two different perspectives (i.e.,optical-flow and pixel image) via flow and image encoders, respectively, for subsequent multi-scale patch generation. Third, the extracted multi-scale patch representations are embedded through the proposed flow-guided and image-guided transformer modules, for spatial-temporal information extraction (Sec. 5.3.1). Fourth, the proposed fusion attention block guides the feature coordination from optical flow and image level during the aggregation which provides a more stable fusion procedure. (Sec. 5.3.2).

Finally, a decoder up-scales the fused features and reconstructs them to a final video sequence, i.e., $\hat{\boldsymbol{Y}}^T = \{\hat{\boldsymbol{y}}_1, ..., \hat{\boldsymbol{y}}_T | \hat{\boldsymbol{y}}_T \in \mathcal{R}^{H \times W \times 3}\}$

Figure. 5.2 presents the whole pipeline of the proposed FDTN framework. It is worth noticing that all modules are differentiable and constitute an end-to-end trainable architecture.

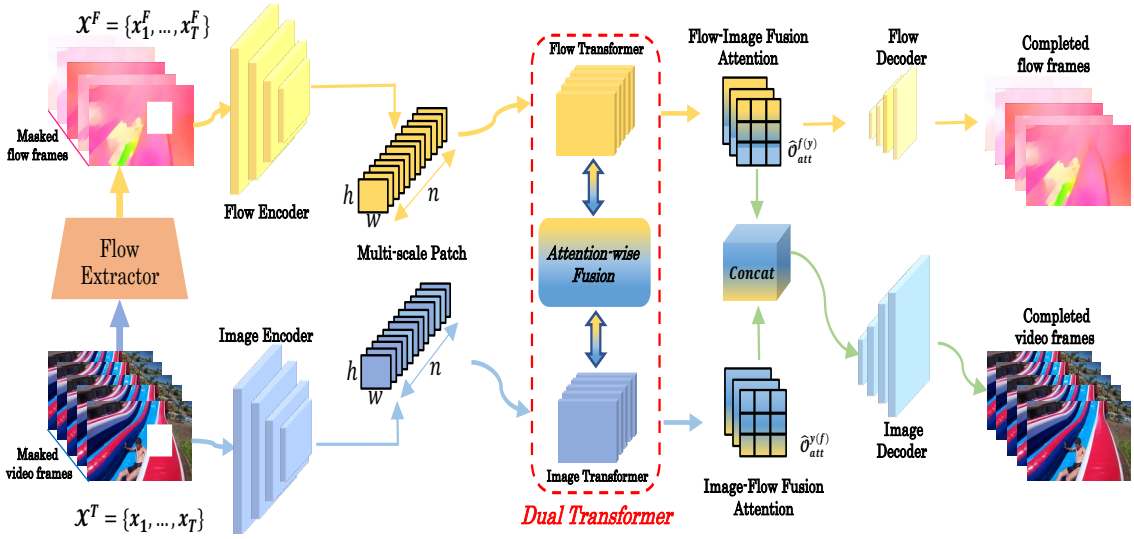


Figure 5.2: Overview of our proposed Flow Enhanced Dual Transformer (FDTN). The model takes masked optical-flow and masked frames as input and extracts multi-scale patches based on two streams. Then the Dual Transformer block takes the multi-modal patches as input and learns fused spatio-temporal representations. Finally, the model generates the completed frame sequence based on the fused representations.

5.3.1 Flow-guided Transformer

"Learning from multi-modalities" are widely studied in various applications [213, 214], which helps to understand and analyze better when various senses are engaged in the processing of information. Optical-flow modality is widely used in the computer vision community which often provides temporal information [48, 210, 211] for mainstream RGB-image tasks. Inspired by that, we proposed a flow-guided transformer module, which aims to extract informative temporal information to encourage image synthesizing. In previous works [215], patch-based transformer methods have been verified for their effectiveness in extracting multi-scale informative representations. Thus, we follow the previous approaches and design a flow-guided transformer

based on multi-scale patches for multi-scale temporal representation extraction.

Specifically, each optical-flow feature encoded by flow encoder, i.e., $\boldsymbol{f}^e \in \boldsymbol{F}_{\boldsymbol{e}}^{\boldsymbol{o}}(\mathcal{X}^F)$, is firstly divided into frame-wise n patches, where $n = \frac{H}{h} \times \frac{W}{w}$ and (h, w) is the patch size. After patching, the entire optical-flow sequence $\boldsymbol{F}_e^p = \{\boldsymbol{f}_{\boldsymbol{i}}^{\boldsymbol{p}} \in \mathcal{R}^{h \times w \times C_f}\}_{i=1}^{T \times n}$ can be fed into the multi-head self-attention layers for spatial-temporal contents extraction.

In the self-attention layer, the Query $\boldsymbol{Q}_f$, Key $\boldsymbol{K}_f$ and Value $\boldsymbol{V}_f$ are encoded through three convolution operators, $E_q(\cdot)$, $E_k(\cdot)$ and $E_v(\cdot)$:

$$\boldsymbol{Q}_f, \boldsymbol{K}_f, \boldsymbol{V}_f = E_q(\boldsymbol{F}_e^p), E_k(\boldsymbol{F}_e^p), E_v(\boldsymbol{F}_e^p). \tag{5.1}$$

Then, the flow-guided transformer aggregates the information based on the optical-flow dot product attention matrix $\mathcal{A}_f$m which is derived through the $\boldsymbol{Q}_f$ and $\boldsymbol{K}_f$:

$$\mathcal{F}_{att}^f(\boldsymbol{Q}_f, \boldsymbol{K}_f) = \frac{\boldsymbol{Q}_f \boldsymbol{K}_f^T}{\sqrt{h \times w \times C_f}}, \tag{5.2}$$

$$\mathcal{A}_f = \frac{exp(\mathcal{F}_{att}^f(\boldsymbol{Q}_f, \boldsymbol{K}_f))}{\sum_{i=1}^{T \times n} exp(\mathcal{F}_{att}^f(\boldsymbol{Q}_f, \boldsymbol{K}_f))}. \tag{5.3}$$

Finally, the output of the optical-flow transformer $\mathcal{O}_f$ can be acquired and described by:

$$\mathcal{O}_f = \mathcal{A}_f \boldsymbol{V}_f. \tag{5.4}$$

Similarly, to aggregate content information from pixel-level images, we utilize an image-based transformer to extract image pixel attention output $\mathcal{O}_t$ and its attention matrix $\mathcal{A}_t$ for subsequent fusion usage.

5.3.2 Dual Transformer with Attention-wise Fusion

To utilize both the content and spatial-temporal information for a more comprehensive image synthesizing, we design a dual transformer structure with multiple attention-wise fusion modules. The proposed attention-wise fusion module (as shown in Figure. 5.3) aims to encourage information complementation from the opti-

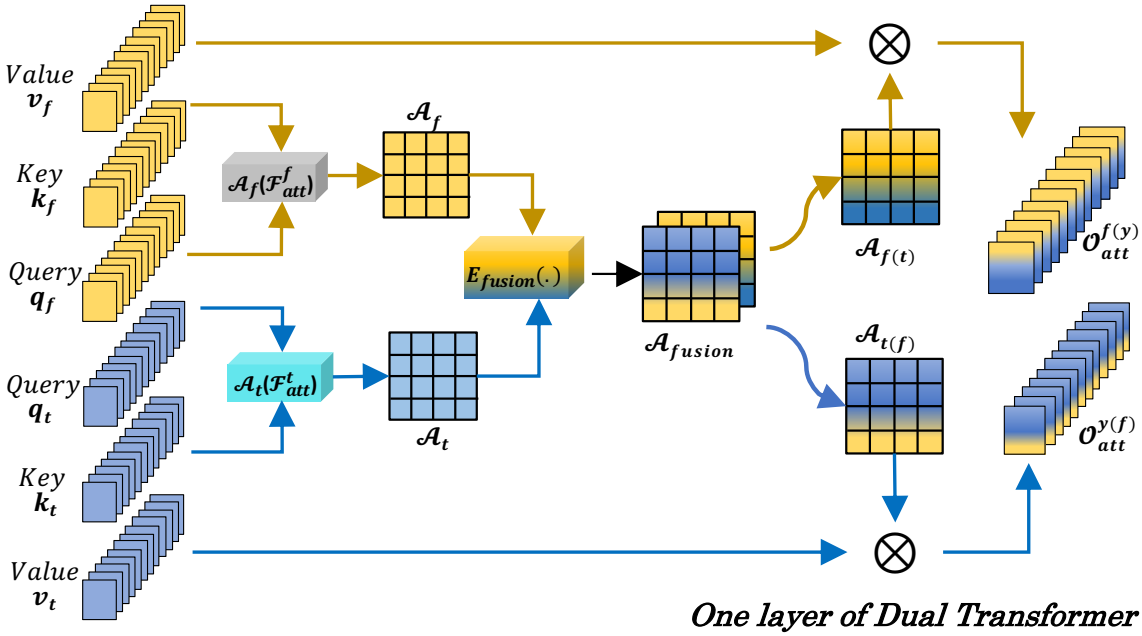


Figure 5.3: The structure of single attention-wise fusion layer.

cal flow attention feature to the pixel image attention feature and vice versa. Firstly, both image feature map and optical flow are extracted by the same image, which can be seemed the same modal. However, image feature and optical flow emphatically represent spatial and temporal information, respectively. Secondly, the proposed attention fusion model executes a fusion operation in the internal attention, meaning that the fusion operation of two features is executed in the internal integration between query, key and value. The attention-wise fusion module first stacks the flow-based and pixel-based attention matrix, i.e., $\mathcal{A}_f$ and $\mathcal{A}_t$, into an image-like tensor with two channels. Then, a fusion convolution operator ($E_{fusion}(\cdot)$) takes the image-like tensor as the input to perform information complementation. A 2D convolution with kernel seize 3 indicates our $E_fusion(.)$ operator. For the split operation, we directly extract the first dimension of fusion output A_{fusion} as the flow-image attention matrix $A_{f(t)}$ and others as image-flow attention matrix $A_{t(f)}$.

The fusion attention matrix $\mathcal{A}_{fusion}$ then can be obtained by:

$$\mathcal{A}_{fusion} = E_{fusion}([\mathcal{A}_f, \mathcal{A}_t]), \tag{5.5}$$

where $[,]$ denotes the stack operation and $\mathcal{A}_{fusion} \in \mathbb{R}^{(T\times n)\times(T\times n)\times 2}$.

Then, the fusion output $\mathcal{A}_{fusion}$ is further separated into two information complementary attention matrices for flow-image attention matrix $\mathcal{A}_{f(t)}$ and image-flow attention matrix $\mathcal{A}_{t(f)}$, respectively. Finally, the flow-image and image-flow attention matrices follow the flow-based transformer attention function to get the attention-wise fusion outputs respectively, which is denoted by:

$$(\mathcal{O}_{att}^{f(t)}, \mathcal{O}_{att}^{t(f)}) = \mathcal{A}_{f(t)}\boldsymbol{V}_f, \mathcal{A}_{t(f)}\boldsymbol{V}_t. \tag{5.6}$$

For multi-layer attention-wise fusion module, the flow-image and image-flow attention outputs from the previous layer are served as input for the next attention-wise fusion layer with the same fusion operation. The final attention-wise fusion layer outputs, $\hat{\mathcal{O}}_{att}^{f(t)}$ and $\hat{\mathcal{O}}_{att}^{t(f)}$, are concatenated for non-corrupted pixel image and optical flow reconstruction.

5.3.3 Fusion Optimization

We utilize the real video frames $\boldsymbol{Y}^T = \{\boldsymbol{y}_1, ..., \boldsymbol{y}_T | \boldsymbol{y}_T \in \mathcal{R}^{H \times W \times 3}\}$ as the supervision information to optimize the proposed FDTN framework through the L_{totoal} objective function:

$$L_{totoal} = L_{in}^t + L_{sr}^t + \beta_{in}^f L_{in}^f + \beta_{sr}^f L_{sr}^f \tag{5.7}$$

where L_{in}^t, L_{sr}^t, L_{in}^f, L_{sr}^f represent the inpainting loss and surrounding loss of pixel image and optical-flow respectively. The β_{in}^f and β_{sr}^f are two hyperparameters that control the contribution index of optical flow modality.

Inpainting Loss and Surrounding Loss

Specifically, the inpainting loss L_{in}^y aims to optimize the corrupted area synthesizing, which calculates the per-pixel restoration accuracy on the corrupted area between the synthesized frame and the ground truth frame:

$$L_{in}^y = \frac{||(\boldsymbol{Y}^T - \mathcal{D}_{image}(\hat{\mathcal{O}}_{att}^{f(t)}, \hat{\mathcal{O}}_{att}^{t(f)})) \odot \boldsymbol{M}^T||_1}{||\boldsymbol{M}^T||_1}. \tag{5.8}$$

where $||\cdot||_1$ indicates the L_1 norm, and $\mathcal{D}_{image}(\cdot)$ represents the pixel image decoder

for reconstruction.

Similarly, the surrounding loss L_{sr}^{y} aims to optimize the non-corrupted area synthesizing, which calculates the per-pixel restoration accuracy on the non-corrupted area between the synthesized frame and the ground truth frame:

$$L_{sr}^{y} = \frac{||(\boldsymbol{Y}^{T} - \mathcal{D}_{image}(\hat{\mathcal{O}}_{att}^{f(t)}, \hat{\mathcal{O}}_{att}^{t(f)})) \odot (1 - \boldsymbol{M}^{T})||_1}{||1 - \boldsymbol{M}^{T}||_1}. \tag{5.9}$$

In the same way, the inpainting loss and surrounding loss for optical flow modality can be written as:

$$L_{in}^{f} = \frac{||(\boldsymbol{F_e}^{T} - \mathcal{D}_{flow}(\hat{\mathcal{O}}_{att}^{t(f)}) \odot \boldsymbol{M}^{T}||_1}{||\boldsymbol{M}^{T}||_1}. \tag{5.10}$$

$$L_{sr}^{f} = \frac{||(\boldsymbol{F_e}^{T} - \mathcal{D}_{flow}(\hat{\mathcal{O}}_{att}^{t(f)}) \odot (1 - \boldsymbol{M}^{T})||_1}{||1 - \boldsymbol{M}^{T}||_1}, \tag{5.11}$$

where $\boldsymbol{F_e}^{T}$ is the ground truth optical-flow images, and $\mathcal{D}_{flow}(\cdot)$ represents the optical flow decoder for reconstruction. Since the texture information from pixel image marginally contributes to the reconstruction of optical flow, we only use the flow-image fusion output ($\hat{\mathcal{O}}_{att}^{t(f)}$) for optical flow reconstruction.

5.4 Experiment

5.4.1 Experiment Setting

Dataset. To evaluate the effectiveness of our FDTN framework, we perform it on two commonly used datasets, namely DAVIS [55] and YouTube-VOS [55] datasets. DAVIS dataset includes 150 high-quality video clips. The testing part has 90 video clips with foreground object masks annotated, and the remaining 60 video clips are used for training. YouTube-VOS covers 4,453 video clips without object mask annotations in total, which are divided into 3,471 video clips for training, 474 video clips for validation, and 508 video clips for testing.

As for masks, during training, we generate stationary and object-like masks to simulate video completion and object removal applications following [193]. For evaluation, stationary masks are used to calculate objective metrics (i.e, quantitative

comparisons), and object-like masks are adopted for qualitative comparisons because of the lack of references.

Metrics. We choose PSNR, SSIM [216] and VFID [193] to evaluate the performance of recent video inpainting methods. Specifically, PSNR and SSIM are widely used for distortion-oriented image and video assessment. VFID has been adopted in recent video inpainting works [193] to measure the perceptual similarity between two input videos.

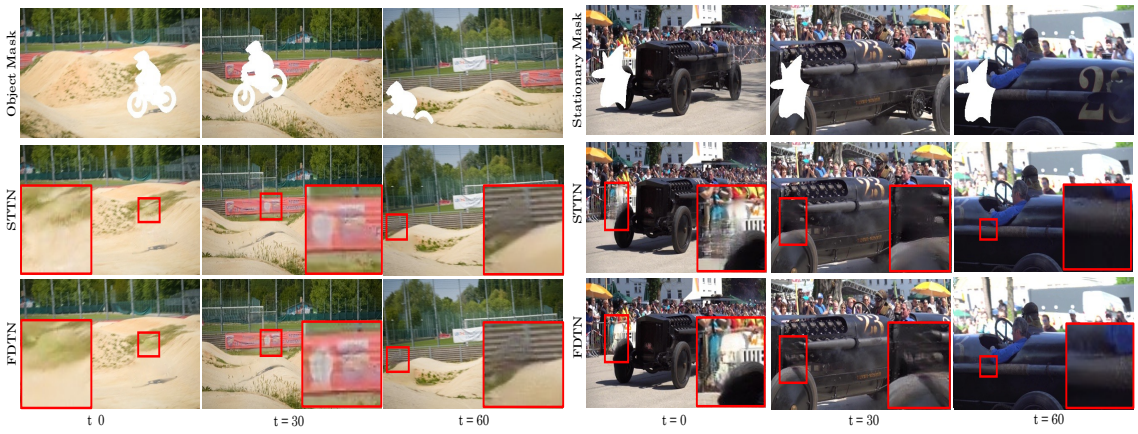


Figure 5.4: The qualitative evaluation comparison between STTN and FDTN based on object mask setting and stationary mask setting.

5.4.2 Performance Comparison

Quantitative Evaluation

We report quantitative results on YouTube-VOS and DAVIS datasets under stationary masks and compare our method with previous video inpainting methods, including VINet [56], DFVI [51], LGTSM [59], FGVC [63] and STTN [193]. As shown in Table 5.1, our method substantially surpass all previous methods on all four metrics. The superior improvements in PSNR and SSIM metrics represent that our method can generate videos with less distortion. The lower results on VFID metric demonstrate that our method can generate videos with more visually plausible content. All of those performance gains verify the superiority of the proposed method.

Table 5.1: The quantitative comparison of between state-the-of-arts based on YouTube-VOS and DAVIS datasets.

Method	YouTube-VOS			DAVIS		
	PSNR	SSIM(%)	VFID	PSNR	SSIM(%)	VFID
VINet [56]	29.20	94.34	0.072	28.96	94.34	0.199
DFVI [51]	29.16	94.29	0.066	28.81	94.04	0.187
LGTSM [59]	29.74	95.04	0.070	28.57	94.09	0.170
FGVC [63]	32.03	95.47	0.063	31.38	95.92	0.143
STTN [193]	32.34	96.55	0.059	30.28	95.21	0.149
FDTN	**33.30**	**96.76**	**0.058**	**33.94**	**96.47**	**0.118**

Qualitative Evaluation

We choose the most representative method STTN [193] to conduct visual comparisons. Figure. 5.4 shows the video inpainting result under stationary masks and object mask settings. For the object mask setting, both STTN and FDTN can synthesize the background to replace the missing object in the foreground, but the generated background of STTN is more blurry compared with FDTN. For the object mask setting, although STTN and FDTN can fill the corrupted area of a black car, FDTN can generate the corrupted area with more faithful textural and structural information. This demonstrates the effectiveness of the proposed method.

5.4.3 Ablation study

To evaluate the effectiveness of our proposed attention-wise fusion module, we conduct an ablation study on the YouTube-VOS dataset. The ablation study consists of three settings: 1) **Early Feature-wise Fusion**: a naive feature-level fusion approach that concatenates the optical flow and pixel image before the transformer module; 2) **Late Feature-wise Fusion**: a feature-level fusion approach that concatenates the attention outputs of optical flow and pixel image after the transformer module; 3) **Attention-wise Fusion**: Our proposed multi-layer attention-wise fusion.

Table 5.2 and Figure 5.5 and Figure 5.6 demonstrate the results for three fusion

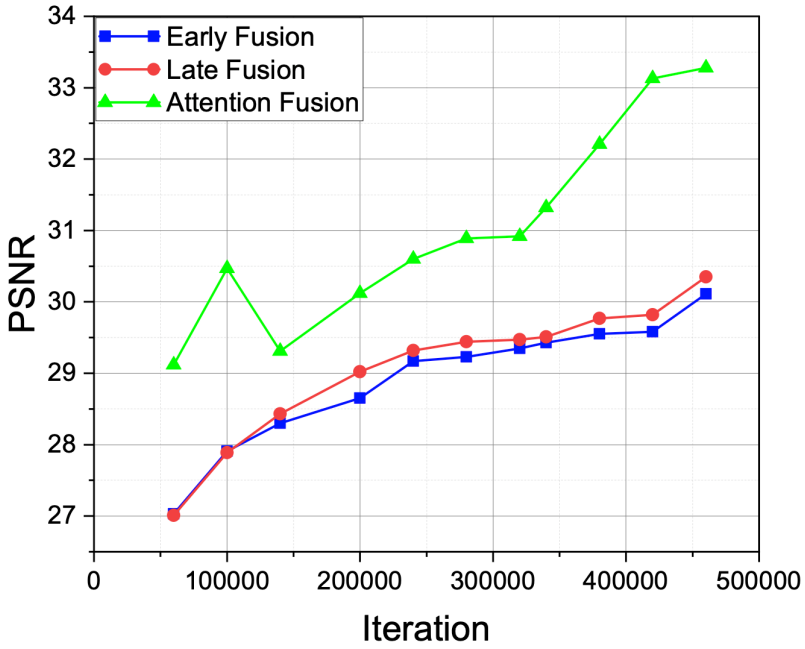
	Evaluation	
Method	PSNR	SSIM(%)
Early Fusion	30.11	91.92
Late Fusion	30.35	92.31
Attention Fusion	**33.30**	**96.76**

Table 5.2: The quantitative comparison based on PSNR and SSIM between fusion methods.

methods. We can find that the attention-wise fusion approach achieves the best evaluation result on PSNR and SSIM metrics, which verifies the effectiveness of the attention-wise fusion mechanism for information complementation.

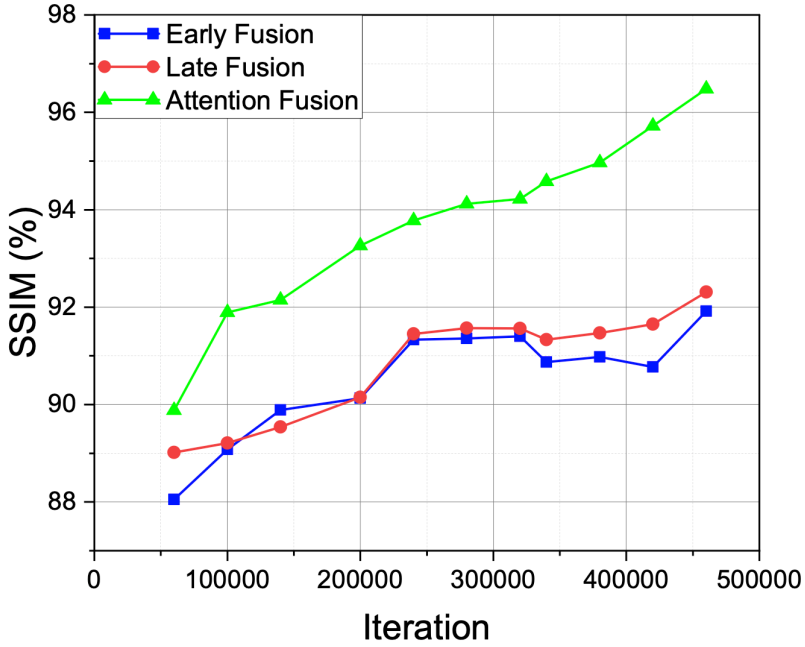
5.5 Conclusion

In this paper, we proposed a novel end-to-end Flow Enhance Dual Transformer Network (FDTN), which explored spatial-temporal knowledge from both image content information and optical-flow motion information. Specifically, the FDTN consists of dual transformer models extracted features from pixel domain and optical flow domain, and connected by the attention-wise fusion module for spatial-temporal information cross-complementation, resulting in more comprehensive image synthesizing. Our FDTN achieved state-of-the-art results on two video inpainting benchmarks YouTube-VOS and DAVIS datasets, which demonstrate the effectiveness of the proposed model.



(a) PSNR

Figure 5.5: The evaluation metrics comparison based on PSNR between different fusion methods.



(b) SSIM

Figure 5.6: The evaluation metrics comparison based on SSIM between different fusion methods.

CHAPTER 6

Sequential Visual-Semantic Analysis with Cycle-based Framework

Image captioning is also a sequential modelling problem, similar to video inpainting, which involves generating a textual description of an image by analysing its content. This makes it a natural extension of video inpainting and can leverage the same types of techniques and approaches, such as recurrent neural networks (RNNs) and attention mechanisms. Secondly, image captioning has practical applications in areas such as image search, automated content tagging, and visual storytelling. By generating descriptive captions for images, image captioning can improve search results and help people with visual impairments understand the content of images. Thirdly, image captioning is a more challenging problem than video inpainting, as it involves not only understanding the content of an image but also generating a coherent and grammatically correct textual description of it. This requires more advanced techniques and models, such as the combination of CNNs and RNNs. Finally, image captioning is an active research area, with new approaches and models being developed constantly, especially in the new AIGC era. By moving from video inpainting to image captioning, we can testify or sequential modelling framework in up-to-date domains with the latest advancements in the field and potentially make

significant contributions to the vision-language area.

Efforts have been made to address the multi-modal task, which includes the classic task of image captioning. The Clip model has been particularly effective in improving the performance of image captioning. However, its few-shot and zero-shot problems have become a significant research focus. In this work, we propose a new few-shot and zero-shot setting for the image captioning task, which differs from popular research directions. Specifically, our approach focuses on the impact of the existing dataset on the captioning model's ability. Our analysis reveals that the frequency of word combinations directly affects the performance of the captioning model. Based on this observation, we define the new few-shot and zero-shot settings. To address this challenge, we propose a Cycle-based captioning framework based on data augmentation, with the novelty switcher module as a critical component. Our experiments demonstrate that our proposed framework achieves state-of-the-art performance on both traditional, few-shot and zero-shot settings.

6.1 Introduction

As a traditional task in deep learning, image captioning aims to describe an image in natural language. Therefore, it generates a sequence of words by designing a model to reflect the relationship between visual and textual information. The caption is the most significant piece of information; it represents a classic example of sequential data. Different word orders can result in distinct descriptions. Therefore, the model deals with sequential information when extracting semantic content from captions. Recently, many efforts paid attention to tackle this task by applying Recurrent Neural Network models [78], Graph Neural Networks [95] and Transformer [109]. Especially the Transformer model can extract more key knowledge between an image and caption to achieve state-of-the-art performance.

As a significant revolution in deep learning, few-shot and zero-shot learning provide more inspiration. For example, the zero-shot capability was demonstrated in computer vision [38]. Besides, the seminal CLIP [112] image-text transformer model can execute tens of downstream tasks without further training. Impressively, the

DALL-E [217] can generate images in terms of unseen descriptions. Although existing algorithms can generate good descriptions on the traditional testing set, the image captioning task needs to be more attentive to the few-shot and zero-shot settings.

In the visual-textual multi-modal task, the relationship between images and texts is the most important factor. Therefore, several efforts have analyzed the impacts of the word on the model performance. For instance, Yan [218] demonstrated that word frequency affects image-text matching model performance. However, compared with a single word, word combinations are more critical to the meaning and understanding of a sentence. Different word combinations include amounts of semantic information, which means that the frequency of word combinations has more effects on model performance compared with the frequency of a single word. Figure 6.1 describes the proportions of the normal, few-shot and zero-shot based on word combination frequency on the test set. The few-shot and zero-shot settings have fewer proportions than the normal setting, which are 9%, 28% and 63%. In order to prove our hypothesis, we do experiments to analyze the impact of word combination on the image captioning task in the methodology section. Furthermore, different from traditional few-shot and zero-shot settings, we propose a new direction of few-shot and zero-shot settings based on this hypothesis in the image captioning task.

Generally, few-shot and zero-shot methods increase the model generalization to improve the model performance in the kinds of tasks, and data augmentation is the most straightforward direction. In terms of this, we propose a novel Cycle Captioning Framework to improve the model ability on the traditional setting and the new few-shot and zero-shot settings for the image captioning task. In the framework, the proposed Image Generator generates the image with feature-level as new training data to feed into the Caption model; meanwhile, the proposed Word Switcher reasonably exchange words of the caption to augment the training data. Summary the contributions:

- According to the analysis impact of word combination on the image captioning task, the new few-shot and zero-shot settings are proposed in this work. While improving the performance of new settings promotes the extension of a new

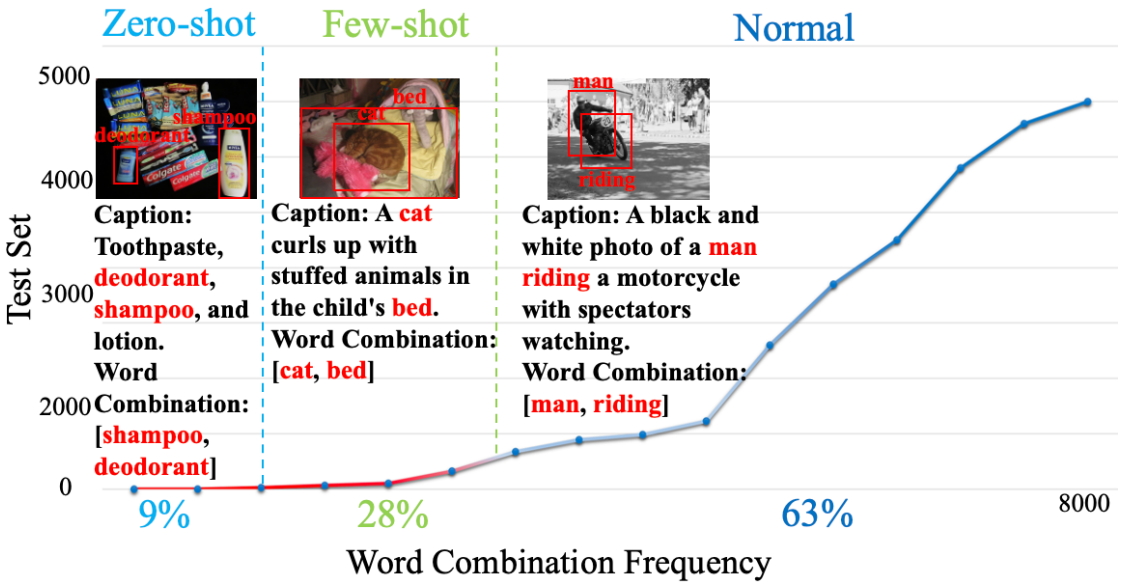


Figure 6.1: The normal, few-shot and zero-shot settings on the Test Set based on Word Combination Frequency.

direction on the image captioning task.

- The proposed Cycle Captioning Framework adequately apply the existing data to improve the model generalization ability on the image captioning task. At the same time, we design a novel Word Switcher to augment the training data.
- The experiments demonstrate that the Cycle Captioning Framework with Word Switcher achieves state-of-the-art performance compared with existing image captioning methods.

The methodology section describes the details of word combination, Cycle Captioning Framework and Word Switcher. The experiment and ablation study sections analyse the ability of the proposed algorithm on the image captioning task.

6.2 Related Work

As a traditional task, many efforts were applied to image captioning. Specifically, Benjamin and J.mao first proposed the deep learning algorithm to predict the sequence of captions in image captioning [80, 219]. With the development of machine learning techniques, more and more works extracted and interacted with the

spatial and relationship semantics between objects based on Attention and Graph neural networks [84, 90, 95, 98, 220]. Subsequently, the image could be extracted more details through a transformer with self-attention to improve the model performance [75, 107, 123, 221]. On the one hand, plenty of efforts solved the text problems based on improvements to the language model such as LSTMs, Transformer, and CNNs [79, 110]. On the other hand, the generated language of image grounding and non-vision words obtained a better performance by combination with different semantic information [109, 122]. Specifically, as a popular language framework, Transformer has been widely used in image captioning tasks. The CogView constructed a 4-billion-parameter Transformer with an image-text tokenizer method to achieve a novel captioning-generating performance [222].

Despite the captioning model experiencing an improvement, also trained the large-scale vision-language data sets can improve the generating captioning performance. Thus, some image captioning tasks applied the large-scale vision-language data sets in recent years, such as the Visual Genome and MS-COCO. The captioning model utilized millions of image and text pairs from the web to improve the generated language performance [223] [224]. Based on this technique, some methods applied the unsupervised external data through conditioning the model during the training to focus on describing novel objects [225] [226]. The model can execute external object information in the pre-training and inference phases [227]. The model can join an image-language embedding space and the visual detector for the unsupervised methods [228] [229].

Following this direction, the zero-shot language model CLIP was proposed, which acquired a better score in the image captioning task based on 400M image-sentence pairs from the web [112]. Based on powerful CLIP, text-driven image manipulation with Generative Adversarial Networks (GANs) and other generative models can be supported by means of CLIP [230] [231]. Furthermore, the Clip-VL model [113] uses the pre-trained Clip model to extract the image region feature.

Unlike existing image captioning few-shot and zero-shot learning directions, we propose new few-shot and zero-shot settings in image captioning. Our framework can augment the image-captions pairs based on an existing data set.

6.3 Methodology

6.3.1 Few-shot and Zero-shot settings

Many machine learning tasks apply the MS-COCO dataset as a traditional dataset, such as object detection, text-image generation and text-image matching. Obvious, the MS-COCO also is the most popular dataset in the image captioning task. Therefore, we analysis the MS-COCO dataset to define our few and zero-shot settings.

As a key part, we define the word combination that two objects or two nouns of a caption construct a word combination; for example, in the caption "A woman is drinking water.", we define the '[woman, water]' to be a word combination. Each image includes five captions in the MS-COCO dataset, and we collect all word combinations from all captions, including about 44,712,680 word combinations.

Based on the word combination, we define the few-shot and zero-shot settings. Firstly, we count the frequency of all word combinations, including the training set and testing set and sort them based on their frequency. Then, applying the SOTA models evaluate the data of high-frequency and low-frequency word combinations, respectively. In this evaluation, the CIDEr and BLEU-4 scores reflect the impacts of high-frequency and low-frequency word combinations on the performance of SOTA models, which is described by Figure 6.2. It shows that the CIDEr and BLEU-4 scores decline with the decrease of the frequency of the word combination, which demonstrates that SOTA models have terrible performance on the data of low-frequency word combinations compared with the data of high-frequency word combinations. Finally, we define the zero-shot test set and the few-shot test set, respectively. The data of low-frequency word combinations of the test set that do not appear in the training set indicates the zero-shot test set. The few-shot test set is the data of low-frequency word combinations of the test set whose appearing frequency in the training set is less than or equal to K.

6.3.2 Cycle Captioning Framework

The definition of the few-shot and zero-shot settings show that the amount of data can directly affect the performance of the captioning model. We propose a cycle

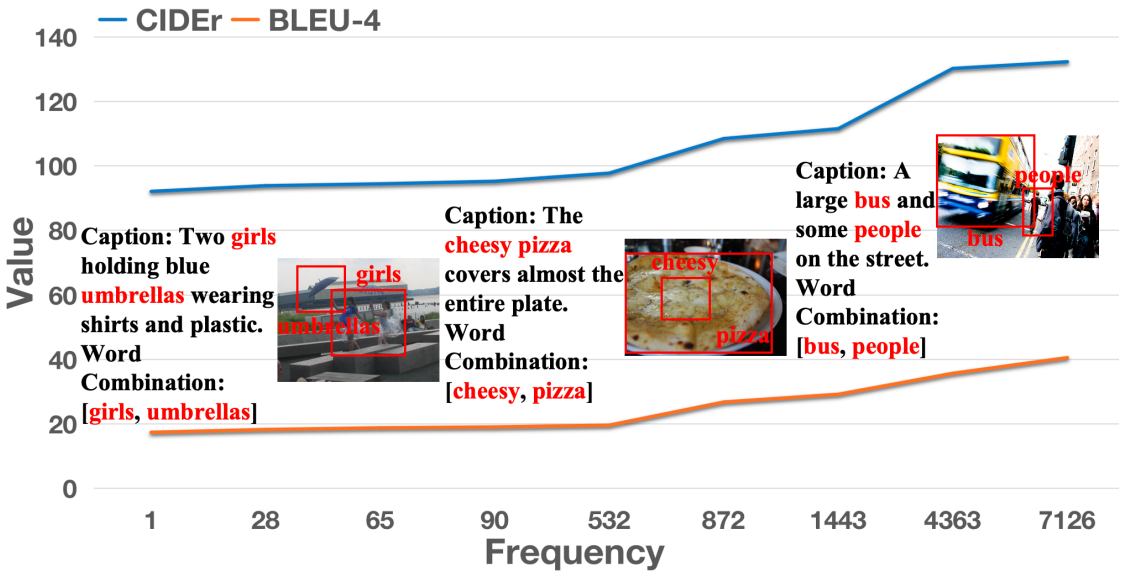


Figure 6.2: The developments of CIDEr and BLEU-4 with frequency of word combination

captioning framework that augments data diversity to overcome the problems in the few-shot and zero-shot settings. Unlike other state-of-the-art captioning models, our framework includes a feature-level image generator and word switcher module in addition to the captioning model. The interaction of the latter two modules enhances the data and thus improves the performance of the captioning model. The details of our framework are further described later in the process of cycle captioning framework section.

Process of Cycle Caption Framework

This part describes the details of the cycle process. Given an image feature $\mathcal{X}$ extracted from an image as input, it is sequentially fed into the Captioning model $\mathcal{G}_c(.)$ and the Feature-Level Image Generator $\mathcal{G}_i(.)$ to generate a sequence of vectors $\widetilde{\mathcal{Y}}$ as a caption:

$$\widetilde{\mathcal{Y}} = \mathcal{G}_c(\mathcal{G}_i(\mathcal{G}_c(\mathcal{X}), \mathcal{X})), \tag{6.1}$$

To describe our framework clearly, we first define some variables. In our framework, two types of features are extracted from images $\mathcal{X}$: image feature map X and region features X_R. The sequence of captions $\mathcal{Y}$ is the same as image features, also described by two representations: original captions Y_r and exchanged captions Y_{ex}.

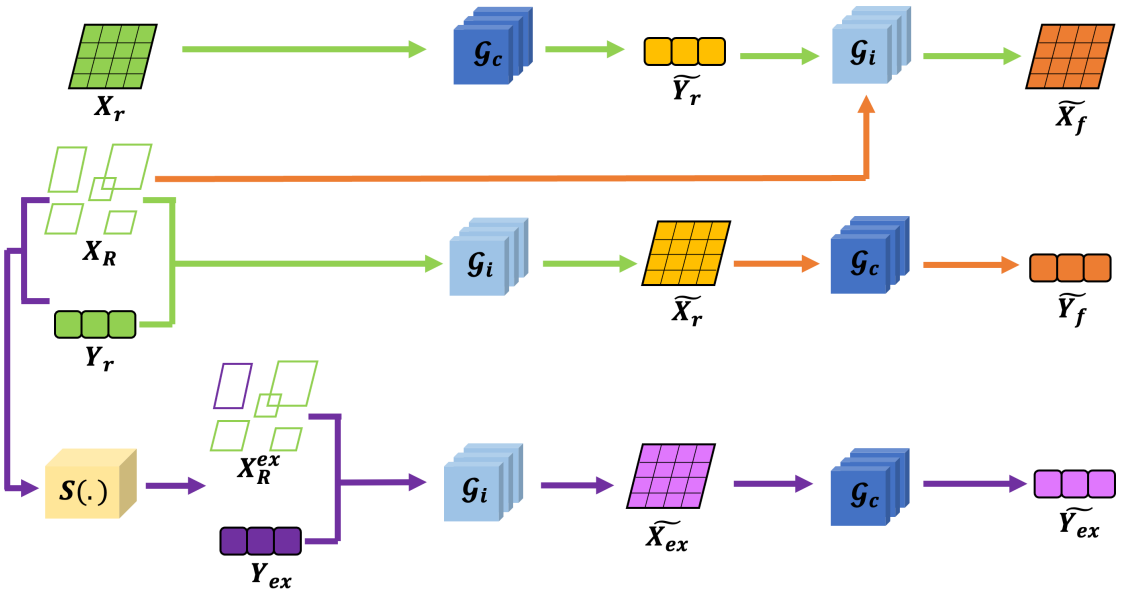


Figure 6.3: The structure of Cycle Captioning framework. The green line is the training process using training data and the orange line indicates the training process using predicted data. The purple line represents the switch module and the training process using exchanged data.

The cycle process of the whole framework is divided into two parts. In the first part, the real caption embeddings Y_r, real image feature map X_r and region features X_R in the training set enter the caption model and feature image generator to obtain the generated caption embeddings $\widetilde{Y}_r$ and image feature map $\widetilde{X}_r$, respectively.

$$\begin{aligned} \widetilde{Y}_r &= \mathcal{G}_c(X_r) \\ \widetilde{X}_r &= \mathcal{G}_i(Y_r, X_R), \end{aligned} \tag{6.2}$$

Then, the generated caption embeddings $\widetilde{Y}_r$ and image feature map $\widetilde{X}_r$ as new training data are fed into two models to acquire cycle caption embeddings $\widetilde{Y}_f$ and cycle image feature map $\widetilde{X}_f$.

$$\begin{aligned} \widetilde{X}_f &= \mathcal{G}_i(\widetilde{Y}_r, X_R) \\ \widetilde{Y}_f &= \mathcal{G}_c(\widetilde{X}_r), \end{aligned} \tag{6.3}$$

Through the above steps, we realized the first step of data expansion without changing the training data so that both models could obtain more data for training.

The second part of the framework is the most critical part of our entire framework. In this part, the caption embeddings and region features X_R in the training set

first input into the word switcher $\mathcal{S}(.)$ to obtain new exchanged caption embeddings Y_{ex} and exchanged region features X_R^{ex}:

$$Y_{ex}, X_R^{ex} = \mathcal{S}(Y_r, X_R), \tag{6.4}$$

Then these new training data are fed into the image generator $\mathcal{G}_i(.)$ and caption model $\mathcal{G}_c(.)$ to generate the predicted exchanged captions $\widetilde{Y}_{ex}$:

$$\widetilde{Y}_{ex} = \mathcal{G}_c(\mathcal{G}_i(Y_{ex}, X_R^{ex}), Y_{ex}), \tag{6.5}$$

Captioning Model

Our caption model, inspired by Mesh-Memory Transformer [109], is represented by $\mathcal{G}_c(.)$. It is the encoder and decoder structure with stacks of self-attention layers. The encoder module extracts the relationships from the input image, and then the decoder module receives the output of the encoder module to predict each word of a caption. All connections between the image and caption are executed by dot-product attention. The attention operator follows the standard sets of the transformer, namely a set of queries Q, keys K and values V, and according to the weighted sum of value vectors with aggregation between query and key vectors. The operator is shown as:

$$Attention(Q, K, V) = softmax(QK^T/d)V, \tag{6.6}$$

where Q is a matrix of n_q query vectors, K and V both contain n_k keys and values, all with the same dimensionality, and d is a scaling factor.

The encoder layers include the self-attention and position-wise feed-forward with a residual connection and a layer norm $Addnorm$, and then stacks of them define our encoder module:

$$O_{ce} = Addnorm(\mathcal{F}(Attention(W_q X_r, W_k XR, W_v X_r))), \tag{6.7}$$

where W_q, W_k, W_v indicate the matrices of learnable weights and $\mathcal{F}(.)$ is position-wise feed-forward layer. The O_{ce} represents the output of encoder module.

Then, the decoder collects outputs from the encoder module and the self-attention mask module S_{mask} to obtain a generated caption $\widetilde{\mathcal{Y}}$, which is described by:

$$\widetilde{\mathcal{Y}} = Addnorm(\mathcal{F}(Attention(O_{ce}, S_{mask}(\mathcal{Y})))), \tag{6.8}$$

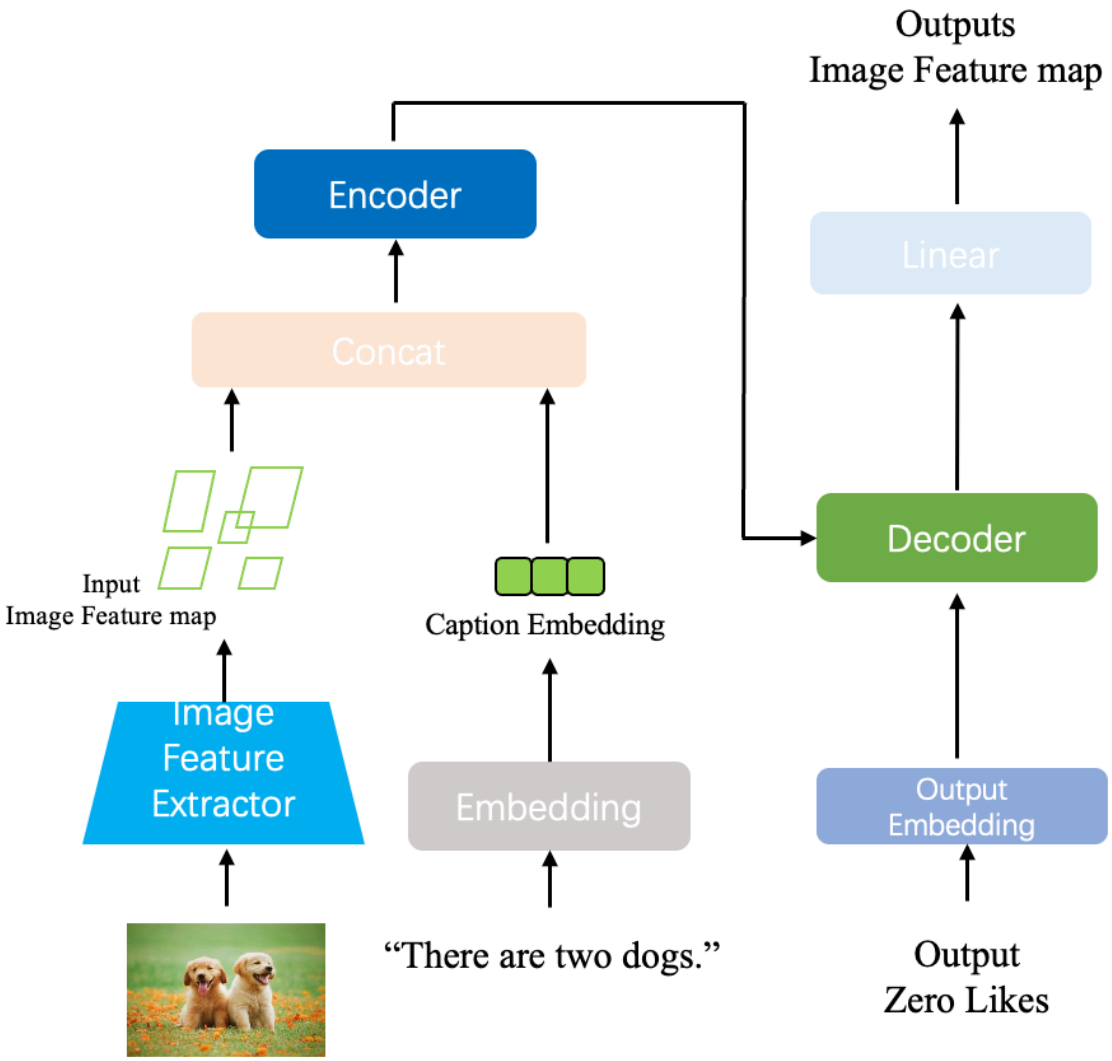


Figure 6.4: The structure or Feature-Level Image Generator

Feature-Level Image Generator

Generally, the image generator synthesis an entire image in the most multi-modal task, which is hard to optimize. However, the proposed feature-level image generator only generates the image features from captions. The $\mathcal{G}_\backslash(.)$ represents the feature-

level image generator (FL image-G) in this work, whose structure is similar to the captioning model based on the transformer. The main difference is that captions $\mathcal{Y}$ combined with the image region feature X_R are the inputs to generate the image feature map $\widetilde{X}_r$.

In this generator, the image region feature provides extra information to improve the accuracy of the synthesised image feature map. Meantime, the switcher module executes the image region feature to generate the new exchanged image region feature as weak ground truth to train the FL image-G. The reason is that the switcher module can create the exchanged captions but cannot generate the exchanged image feature map, which means that when we apply the exchanged captions to train the FL image-G, there is no ground truth of the image feature map to supervise. But we can directly exchange the region proposal feature corresponding to the exchanged object word to obtain weak ground truth. The next part can describe the details.

Switcher Module

The main novel part in our framework, the word switcher module plays a key role. It is represented by $\mathcal{S}(.)$. We follow a principle every time we change words: we only exchange one noun in a caption. However, the exchanged word is not random because some new captions constructed by newly exchanged words are not reasonable, which means that these data can affect our model performance. Therefore, we follow two steps to select the exchanged word. The first step is choosing the newly exchanged word from our word combination. For example, in the caption "A man plays football.", the word combination is '[football, man]'. The Figure 6.5 describes the process of word combination. $E_w(.)$ extracted four words to four word embedding t_1, t_2, t_3 and t_4. Then $F_{ty}(.)$ as word type filter selects nouns from four word embedding to construct word combination.

We first decide to exchange the word man, and we will select the newly exchanged word from the combination list containing the word 'football', such as '[football, woman]', '[football, cups]' and '[football, dog]' etc. Then, these newly exchanged words construct different new captions: "A woman plays football.", "A cups plays football." and "A dog plays football.". The second step is to compute the similarity

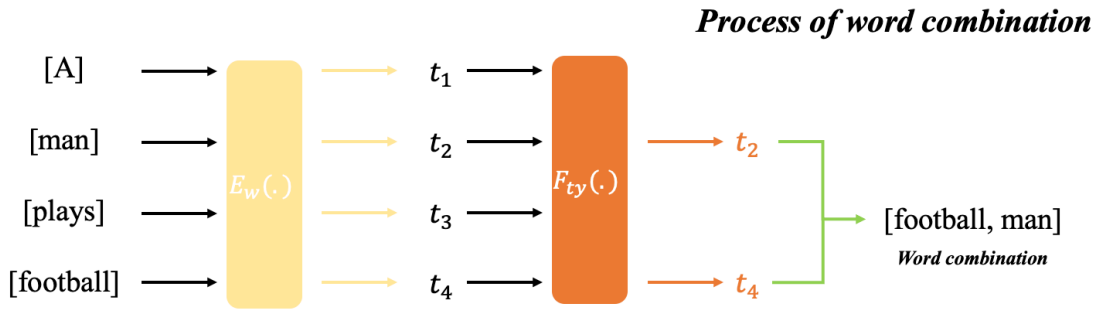


Figure 6.5: Process of word combination

and distance between these new sentences and original sentence to select the final exchanged sentence:

$$\begin{aligned} L_{dis} &= ||Y_o - \dot{Y}_{ex}||_2, \\ L_{sim} &= \frac{Y_o \cdot \dot{Y}_{ex}}{||Y_o||||\dot{Y}_{ex}||} \end{aligned} \tag{6.9}$$

where Y_o and $\dot{Y}_{ex}$ denote the original sentence and the new sentence candidates, L_{dis} and L_{sim} are Euclidean distance and Cosine similarity.

We can obtain the weak exchanged image region feature when we acquire the final newly exchanged caption. Firstly, we collect the representations of each objects in the whole dataset. Then, we obtain the comprehensive representations through the Equation 6.10:

$$r_c = (\sum_{n=0}^{N} r_n)/N, \tag{6.10}$$

where r_c and r_n are the comprehensive representation and each representation of object. For example, if the exchanged word is 'man' and the new word is 'woman', we can acquire their representations from the image region features by class probability. We directly apply the comprehensive representation of 'woman' to replace 'man', which obtains the new exchanged image region feature X_R^{ex} based on this principle. Figure 6.6 describes details.

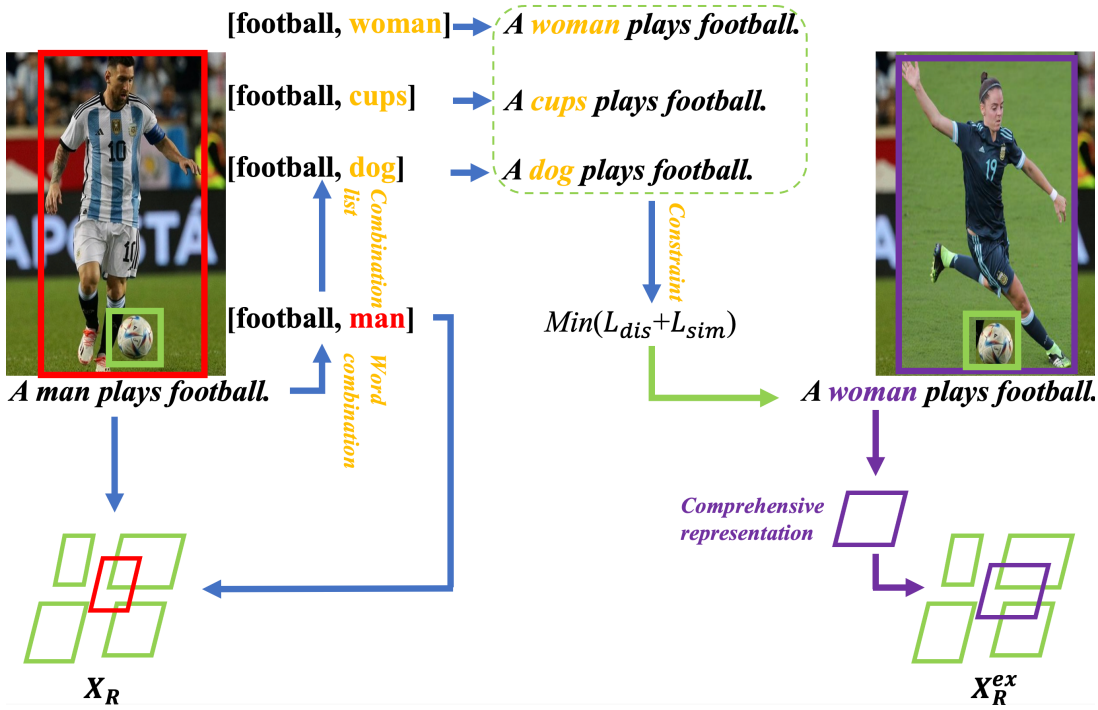


Figure 6.6: The details of the switcher module. The red word is the exchanged word and purple is the new word.

6.4 Experiments

In this section, two evaluation settings demonstrate our model performance. First, our model and state-of-the-art (SOTA) models are evaluated in a traditional setting. Then, our novel few-shot and zero-shot setting is the second setting to evaluate our model and SOTA models. Finally, the test results of our model are compared with the results of the SOTA models under both settings to demonstrate that our model can achieve SOTA performance.

6.4.1 Datasets

The MS-COCO, the most common dataset for image captioning tasks, is applied to evaluate our model performance. The dataset includes more than 120000 images, and 5 different captions annotate each image. Most image captioning tasks widely follow Karpathy's split setup [78], where 110000 images are applied for training, 5000 for validation and the rest for testing.

Regarding the Methodology section, the zero-shot and few-shot settings are set

up based on our word combination principle. Hence, the zero-shot setting splits conventional images for training and validation. We select partial images from the standard test setting for zero-shot testing based on our zero-shot principle. The training and validation set of our few-shot setting also follows the common training and validation setting, and we set K-shot ($K = 2$) to choose testing images.

6.4.2 Experiments Setting

To show our model performance, we follow the standard evaluation protocol to apply the typical image captioning metrics: BLEU [232], METEOR [233], ROUGE [234] and CIDEr [235].

In terms of our framework, an object in the caption is selected randomly to be exchanged with another different object constructing a new caption and then generating a new image feature, which means that the exchanged object of the caption should correspond to the object of images. Hence, we need to obtain image regions in our framework besides the feature map. To acquire image regions, we execute Faster R-CNN [91] with ResNet-101 [85] fine-tuned on the Visual Genome [92] [90] to obtain a 2048-dimensional feature for each region. For caption representation, we linearly project words of one-hot vectors to the input dimensionality of the model d. Then, the positional encoder [75] represents word positions added into the sequence to acquire two embeddings. In our framework, the dimensionality d of each layer is set to 512, the number of memory vectors is 10, and the number of heads is 6. We follow the most common training strategy in image captioning tasks, which is divided into two stages. The first stage is training our captioning model and image generator with a batch size of 256 and learning rate scheduling strategy with a warmup to 100 epochs. Then, two models are optimized with the Adam optimizer, and the beam size is set to 5. The second stage is that the captioning model is fine-tuned with CIDEr-D optimization with a fixed learning rate of 3×10^{-4}.

Table 6.1: The comparison with SOTA on Traditional Setting. B@1, B@4, M, R AND C INDICATE BLEU-1, BLEU-4, METEOR, ROUGE AND CIDER

Method	***Metrics***				
	B@1	B@4	M	R	C
SCST [84]	-	34.2	26.7	55.7	114
Up-Down [90]	79.8	36.3	27.7	56.9	120.1
RFNet [236]	79.1	36.5	27.7	57.3	121.9
GCN-LSTM [95]	80.5	38.2	28.5	58.3	127.6
ORT [123]	80.5	38.6	28.7	58.4	128.3
AoANet [107]	80.2	38.9	29.2	58.8	129.3
M^2 Transformer [109]	80.8	39.1	29.2	58.6	131.2
Clip-VL [113]	-	40.2	31.1	-	134.2
Ours	80.8	***40.6***	***31.6***	***59.3***	***134.6***

6.4.3 Comparison with state-of-the-art methods

In this part, a comparison between the performance of several recent SOTA proposals and our image captioning framework in both settings demonstrates that our framework can achieve SOTA performance. The compared models include SCST [84] and Up-down [90], which applied attention to the grid of features and regions, respectively. Then, the RFNet [236] applies a recurrent fusion network to merge CNN features, and GCN-LSTM [95] executes a Graph CNN to obtain pairwise relationships between image regions. Further, our framework compares with AoANet [107], ORT [123] and M^2 Transformer [109], which apply Transformer for encoding image regions. Finally, we compare with the Clip-VL model [113], which uses the pre-trained Clip model to extract the image region feature. Our framework and aforementioned SOTA models evaluate the traditional test split. Table 6.1 reports the comparison performances, applying the caption model and fine-tuning optimization on the CIDEr score. According to observation from Table 6.1, our framework achieves the best performance on BLEU-1, BLEu-4, METEOR, ROUGE and CIDEr. Our framework especially increases the SOTA on ROUGE by 0.7 points.

Table 6.2: The comparison with SOTA on Few-shot and Zero-shot Setting.

Method	***Few-shot Setting***					***Zeo-shot Setting***				
	B@1	B@4	M	R	C	B@1	B@4	M	R	C
Clip-VL	72.80	19.30	27.38	54.13	97.11	72.27	17.06	26.84	55.30	92.75
Ours	***74.41***	***24.26***	***27.76***	***58.02***	***113***	***73.30***	***22.88***	***29.43***	***58.54***	***110.68***

Because the Clip-VL is the best performance, we compare the testing results with it on our few-shot setting and zero-shot test setting, which are represented by Table 6.2. In particular, we mainly report the performances of the few-shot setting with $K = 2$. As it can be observed from Table 6.2, the performances of all metrics are worse than the traditional test setting, which also proves that the frequency of the word combination can impact the model performance. However, Table 6.2 indicates that our framework surpasses SOTA approach in terms of BLEU-1, BLEU-4, METEOR and ROUGE being the best performer. To further prove our framework performance, Figure 6.7 proposes qualitative results and visualization. In all SOTA approaches, the Clip-VL model is the best performer. Hence, our framework compares with it. On average, our framework can generate more accurate and reasonable captions to describe the corresponding images. In addition, our framework also describes more details and object relationships for images.

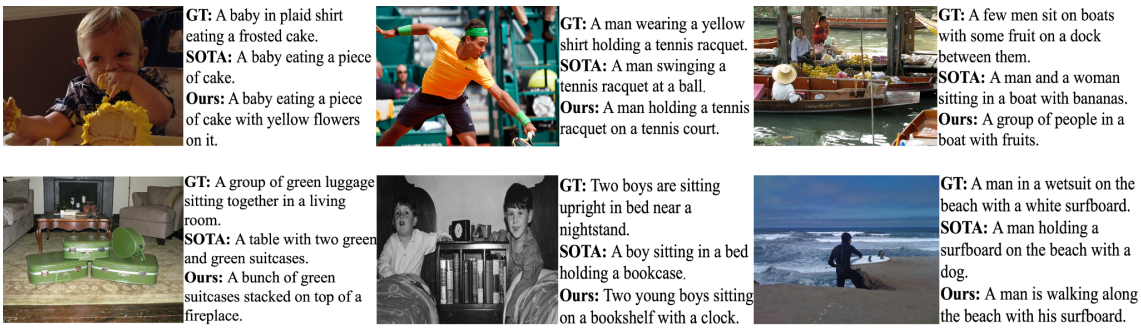


Figure 6.7: The comparison of visualization with SOTA.

In addition, we compare the performance of the single captioning model between our framework and the SOTA model, which is shown by Table 6.3. Because most

Table 6.3: The comparison with SOTA on single captioning model.

Method	***Metric***				
	BLEU-1	BLEU-4	METEOR	ROUGE	CIDEr
Clip-VL	75.30	33.39	27.69	56.09	111.5
Ours	***75.51***	***34.93***	***29.84***	***58.20***	***114.49***

SOTA image captioning models fine-tune the captioning model with an reinforced strategy to improve performance, but a single captioning model is the most significant part, which directly represents the actual ability of captioning generation for each SOTA approach. Table 6.3 reports the results, showing that our framework is the best performer on all evaluation metrics and reflects our framework's superiority. Although our captioning model includes an image generator, it is a crucial augmentation part of our captioning model, and our framework's total number of layers is fewer than other methods.

6.5 Ablation Study

The quantitative and Qualitative results evident that our framework achieves the best performance compared with other SOTA models. Furthermore, this ablation study section reports the effects of the feature-level image generator and switcher module on task performance. To directly analyze the effects of each component, the following experiments are executed based on the single captioning model without the reinforced fine-tuning strategy.

6.5.1 The Effect of Feature-Level Image Generator

A part of the cycle-captioning model, Table 6.5 proves that the feature-level image generator produces essential effects for the whole framework. Compared with the baseline, the entire framework obtains a noticeable improvement in all settings when applying the feature-level image generator. Significantly, the ROUGE increase by approximately 0.51 points compared with the baseline and acquires the best performance on the few-shot setting. Impressively, the CIDEr improves by about 11.29

Table 6.4: The comparisons between different switcher methods.

Method	***Component***			***Traditional Setting***		***Few-shot Setting***		***Zero-stho Setting***	
	Switch	Constraint	Combination list	B@4	C	B@4	C	B@4	C
Without Switch	×	×	×	33.62	110.63	23.08	110.50	22.27	110.02
Random	✓	×	×	33.41	104.38	22.04	104.01	20.66	100.68
Word nearly	✓	✓	×	32.69	104.51	21.09	104.35	22.13	104.23
Combination	✓	✓	✓	***34.93***	***114.49***	***24.26***	***113***	***22.88***	***110.68***

points in the zero-shot setting.

In our cycle framework, the feature-level image generator based on transformer executes the image region feature to generate the image feature map. Besides, we also applied the traditional GAN to model it without the image region feature. The Table 6.5 reports the differences between two methods.

On the one hand, besides the BLEU-4 of the traditional setting, the results of GAN are worse than the transformer with region feature. However, the performance of GAN is improved compared with the baseline on all settings, which further proves that our cycle framework can obtain an enhancement for the image captioning task. On the other hand, the GAN-based image generator only applies the caption to generate the image feature map without any other data to supervise the model further. But the transformer-based generator weakly supervises the model by using the region feature and the caption.

6.5.2 The Effect of Switcher Module

Switcher module is the most novelty component in the whole cycle framework, which can exchange the word of a caption to augment the new training data. Therefore, we try different methods to improve the performance of the framework. All methods are applied based on the cycle framework.

Table 6.4, the Transformer based indicates no switcher module, and Random represents that the switcher module randomly exchanges a word in a caption. Both Word nearly and Combination methods follow the word combination. The first one means that we fix the first word of the word combination and exchange its neighbour;

for example, in a caption "A man plays football.", the word combination is '[man, football]', we will exchange 'man' neighbour 'plays' to other word based on constrain. The second one is that we exchange the second word of the word combination and select a new word from the combination list of the first word; for instance, in a caption "A man plays football.", the word combination is '[man, football]', and we exchange the 'football'. If the combination list may include '[man, tennis]' and '[man, baseball]', we select 'tennis' or 'baseball' to construct a new caption based on the constraint.

Table 6.5: The comparisons between different image generators.

	Traditional Setting		***Few-shot Setting***		***Zero-shot Setting***	
Method	B@4	C	B@4	C	B@4	C
Baseline	32.96	104.22	22.51	110.48	22.19	98.73
GAN based	***33.97***	109.74	22.81	110.48	22.11	107.60
Transformer based	33.62	***110.63***	***23.08***	***110.50***	***22.27***	***110.02***

Table 6.4 indicates that the unreasonable switcher method can destroy the ability of the whole framework, such as the Random method and the Word nearly method. These methods generate the new training data as noise to attack the model. Although the Word nearly method executes the word combination and constraint to generate the new weakly reasonable caption, it still exits the instability when exchanging a near word. Finally, we apply the interaction between the combination list and constraint to generate the new caption as much as stable. Table 6.4 demonstrates that our switcher method supports the framework to achieve the best performance, especially on the few-shot and zero-shot settings.

6.6 Conclusion

In this work, we define the new few-shot and zero-shot settings based on the principle of the word combination. Meanwhile, a cycle-based captioning framework is proposed to solve this task. Firstly, the word combination is designed through the popular dataset. Then, the experiments demonstrate that the word combination frequency can impact the captioning performance of the model, proving that the proposed few-shot and zero-shot settings are reasonable existing. Finally, the cycle-based captioning framework augments the data with a feature-level image generator and the novelty switcher module to achieve state-of-the-art performance on traditional, few-shot and zero-shot settings. Although the cycle-based captioning framework acquires the best ability, the algorithm of the switcher module can still be improved. In the future, we can apply reinforcement learning to design the switcher module, and the reward, as the feedback, can weakly supervise the feature-level image generator.

CHAPTER 7

Conclusion and Future work

This thesis delves into the complexities of sequential information and presents innovative solutions to three major problems: control, vision, and language. Chapter 3 focused on control and proposes two novel reinforcement learning algorithms to effectively manage two different sequential batch processes that are commonly used in mainstream applications. Building upon the analysis of Chapter 3, Chapter 4 introduced a reinforcement virtual learning framework to optimise the sequential batch process further.

In Chapter 5, we shifted our focus to sequential visual information and the challenges associated with it. To address these challenges, we proposed a dual fusion transformer model that effectively captures the spatiotemporal features of visual information and leverages them to enhance performance in various applications.

Finally, in Chapter 6, we address the control problems of sequential visual-semantic information in few-shot and zero-shot settings. To tackle these issues, we propose a cycle-based framework that optimises the interaction between visual and semantic information, resulting in improved performance and greater efficiency. Overall, this thesis presents a comprehensive analysis of sequential information and offers practical solutions to some of its most significant challenges. Key findings of

this thesis are summarised as follows.

7.1 Sequential Process Control by Reinforcement Learning

In chapter 3, we proposed two improved reinforcement learning to control batch processes. Firstly, we aim at different batch processes to design the agent, state, action and reward functions of reinforcement learning. Secondly, we improved the exploration strategy for all actions, leading to efficient learning progress. Additionally, we explored the same action to execute in multiple steps at different periods, inspired by multiple-step action reinforcement learning (MSA). Due to these improvements, we demonstrated that the improved reinforcement learning achieves a state-of-the-art control performance compared to conventional methods.

7.2 Sequential Process Control by Hybrid Reinforcement Virtual Learning

In chapter 4, we proposed a hybrid reinforcement virtual learning framework to optimize process control further. Although improved reinforcement learning achieves a good control performance with fewer experiments in the batch process, the approach still widely relies on the real environment to learn control strategy. Hence, a high-efficiency control strategy is essential in the real industrial control process. Hybrid reinforcement virtual learning contains virtual space and reinforcement learning control part. Specifically, virtual space estimates a virtual environment based on the real environment first. Then, the reinforcement learning control part acquires a virtual control strategy by interaction with virtual space. Because of the virtual control strategy, the reinforcement learning control part with prior knowledge interacts with the real environment to efficiently obtain a real control strategy. Due to virtual space, we can predict future control results under interaction with the virtual environment to adjust the virtual control strategy. An optimal virtual con-

trol strategy guides the control part to optimize the real control strategy during interaction with the real environment. Compared with other control approaches, hybrid reinforcement virtual learning achieves the best control results with fewer interactions with the real environment.

7.3 Sequential Visual Information

In chapter 5, we proposed a dual fusion transformer model to optimize the control challenges in sequential visual information. Firstly, we applied the optical flow to provide extra-temporal knowledge, improving time-series consistency for the video inpainting task. Then, we fusion optical flow features with conventional spatial features by the proposed attention-wised fusion operator to control reasonable contents completing the missing regions of a video sequence. Due to attention-wised fusion operation, temporal knowledge from the optical flow can interact with spatial information from image pixel features to obtain complementary attention clustering outputs, improving the model control of reasonable pixels to complete the target regions sequentially. Unlike other novelty transformer-based models, our attention-wised operator fusions two perspectives in the internal attention integrating operation of the transformer. During this period, two views can provide extra knowledge to support each other at the initial attention integration between query and key. Finally, we executed standard experiments to prove our proposed model achieves state-of-the-art performance compared with different algorithms.

7.4 Sequential Visual-semantic information

In chapter 6, we proposed a cycle-based transformer framework to control sequential visual-semantic information. Specifically, the proposed cycle-based transformer framework improved the performance of the image captioning task aiming at new few-shot and zero-shot settings. Firstly, we designed a feature-level image generator to synthesize image features besides the standard captioning generator. In addition to training two generators with actual training data, the generated image feature and

caption from two generators as augmented training data also train two generators, representing the first data augmentation. Secondly, the proposed switcher module exchanges the object word of a caption for acquiring new caption data and the new estimated image feature, which is the most novelty part of the entire framework and indicates the second data augmentation. According to the proposed feature-level image generator and switcher module, we conduct two data augmentation to directly improve that model control the generated word constructing a sentence satisfying syntactic. Besides, we proposed the new few-shot and zero-shot sets based on the analysis of the conventional dataset. Finally, we execute experiments on the popular test set, new few-shot set and zero-shot set, which demonstrate that the proposed cycle-based framework outperforms state-of-the-art performance under all settings.

7.5 Future Work

Sequential Process Control with Hybrid Reinforcement Virtual Learning In the future, we will apply reinforcement virtual learning (RVL) to real industrial processes. We further optimize the RVL to achieve high-efficiency control performance based on the real environment. For example, fermentation is the most key part in the production process of Penicillin. In the entire fermentation process, pH value, temperature, and oxygen levels significantly impact the quality of the production. We apply reinforcement learning to control fermentation process.The actions are pH value, temperature, and oxygen, while the state is represented by the quality of production. The reward can be reflected by the difference in the quality of production at different steps. Based on this design, we can construct a Q-table to learn a control strategy. In addition, we execute the RVL to control the industrial processes of multiple agents. Because of the flexible framework of RVl, we can use MARL to replace Q-learning. Meanwhile, optimizing virtual space is another essential operation. Due to the decisions of the previous time generating the effects for the late period, we apply an attention-based transformer model to augment the relationships of the entire process.

Sequential Visual Information Analysis In chapter 4, we proposed a dual fu-

sion transformer model based on optical flow to improve the performance of image pixel generation. Due to the importance of optical flow in the whole model, it is essential that the proposed model further optimizes the generation of optical flow based on an attention-wised fusion operator in the future. In addition, we will apply the proposed model to other visual information control, such as video generation and action recognition. Despite the proposed model achieving state-of-the-art performance in video inpainting, the efficiency of the model still is a considerable issue. Hence, the distillation of the model is our future direction. Because we fusion the features of two perspectives in each layer in the proposed model, we can observe and obtain the fixed layer of the optimal fusion result. Based on this, we further distil the model. In the future, we will apply the proposed model in Medical Image Processing. In this application, images contain a significant amount of noise and corruption that affects the semantic quality of image. The proposed method aims to restore and complete the missing regions of the image, resulting in a high-quality medical image.

Sequential Visual-semantic Information Analysis In chapter 5, we proposed a cycle-based captioning framework to address the new few-shot and zero-shot settings problems and achieve state-of-the-art performance. In the future, there will be two main optimization directions. Firstly, we will optimize the feature-level image generator inspired by the diffusion model. Secondly, we can apply reinforcement learning to design the switcher module, and the reward, as the feedback, can weakly supervise the feature-level image generator. Besides, we will execute the proposed cycle-based framework in text-image retrieval and text-image generation tasks. In a real-world application, we will implement the proposed model in industrial intelligent manufacturing. This model will be trained using industrial data. When production exhibits appearance defects, the model will generate descriptions for engineers, thereby improving overall working efficiency

Application of AI technology in Real Industry With the development of AI theories, more and more works paid attention to real industry. AI technology can further change the working process of industry. For example, computer vision technology is applied in industrial defect inspection. The quality of production is

important with the demands of modern marketing. Compared with the traditional detection method, computer vision detection improves detection accuracy. Meanwhile, computer vision detection technology instead of labour decreases the cost. However, the application of deep learning still has some limitations. Firstly, the data of real industrial appearance defects has various types, and quantity is less compared with other tasks. Secondly, the training time of the model increases the production cycle of the product. Hence, we will apply the generation model to augment training data. Meantime, we execute few-shot and zero-shot learning to solve the few and zero data problems. Finally, we further train a large model aiming at industrial defect inspection based on the diffusion model.

Bibliography

[1] W. Bank, "World development report 2021: Data for better lives," 2021. 1

[2] G. Stephanopoulos, *Chemical process control*, vol. 2. Prentice hall Englewood Cliffs, NJ, 1984. 1

[3] R. A. Jarvis, "A perspective on range finding techniques for computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 2, pp. 122–139, 1983. 1

[4] G. Saridis, "Intelligent robotic control," *IEEE Transactions on Automatic Control*, vol. 28, no. 5, pp. 547–557, 1983. 1

[5] M. Pienemann, *Language processing and second language development*, vol. 10. Amsterdam: John Benjamins, 1998. 1

[6] W. S. McCulloch and W. Pitts, "A logical calculus of the ideas immanent in nervous activity," *The bulletin of mathematical biophysics*, vol. 5, pp. 115–133, 1943. 1

[7] H. Yoo, H. E. Byun, D. Han, and J. H. Lee, "Reinforcement learning for batch process control: Review and perspectives," *Annual Reviews in Control*, vol. 52, pp. 108–119, 2021. 1.1, 1.1, 3.1

[8] L. Mears, S. M. Stocks, G. Sin, and K. V. Gernaey, "A review of control strategies for manipulating the feed rate in fed-batch fermentation processes," *Journal of biotechnology*, vol. 245, pp. 34–46, 2017. 1.1

[9] A. Voulodimos, N. Doulamis, A. Doulamis, E. Protopapadakis, *et al.*, "Deep learning for computer vision: A brief review," *Computational intelligence and neuroscience*, vol. 2018, 2018. 1.2

[10] D. H. Hubel and T. N. Wiesel, "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," *The Journal of physiology*, vol. 160, no. 1, p. 106, 1962. 1.2

[11] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological cybernetics*, vol. 36, no. 4, pp. 193–202, 1980. 1.2

[12] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. 1.2

[13] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989. 1.2

[14] M. Tygert, J. Bruna, S. Chintala, Y. LeCun, S. Piantino, and A. Szlam, "A mathematical motivation for complex-valued convolutional networks," *Neural computation*, vol. 28, no. 5, pp. 815–825, 2016. 1.2

[15] Y. Wang, "Survey on deep multi-modal data analytics: Collaboration, rivalry, and fusion," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 17, no. 1s, pp. 1–25, 2021. 1.2, 1.3

[16] J. Summaira, X. Li, A. M. Shoib, and J. Abdul, "A review on methods and applications in multimodal deep learning," *arXiv preprint arXiv:2202.09195*, 2022. 1.2

[17] C. Xu, D. Tao, and C. Xu, "Large-margin multi-viewinformation bottleneck," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 8, pp. 1559–1572, 2014. 1.3

[18] K. Wang, Q. Yin, W. Wang, S. Wu, and L. Wang, "A comprehensive survey on cross-modal retrieval," *arXiv preprint arXiv:1607.06215*, 2016. 1.3

[19] M. Z. Hossain, F. Sohel, M. F. Shiratuddin, and H. Laga, "A comprehensive survey of deep learning for image captioning," *ACM Computing Surveys (CsUR)*, vol. 51, no. 6, pp. 1–36, 2019. 1.3, 2.3

[20] I. Y. Smets, J. E. Claes, E. J. November, G. P. Bastin, and J. F. Van Impe, "Optimal adaptive control of (bio) chemical reactors: past, present and future," *Journal of process control*, vol. 14, no. 7, pp. 795–805, 2004. 2.1.1

[21] M. A. Hussain and K. Ramachandran, "Comparative evaluation of various control schemes for fed-batch fermentation," *Bioprocess and biosystems engineering*, vol. 24, pp. 309–318, 2002. 2.1.1

[22] S. Duan, Z. Shi, H. Feng, Z. Duan, and Z. Mao, "An on-line adaptive control based on do/ph measurements and ann pattern recognition model for fed-batch cultivation," *Biochemical Engineering Journal*, vol. 30, no. 1, pp. 88–96, 2006. 2.1.1

[23] M. Jenzsch, S. Gnoth, M. Kleinschmidt, R. Simutis, and A. Lübbert, "Improving the batch-to-batch reproducibility in microbial cultures during recombinant protein production by guiding the process along a predefined total

biomass profile," *Bioprocess and biosystems engineering*, vol. 29, pp. 315–321, 2006. 2.1.1

[24] R. Oliveira, R. Simutis, and S. F. De Azevedo, "Design of a stable adaptive controller for driving aerobic fermentation processes near maximum oxygen transfer capacity," *Journal of Process Control*, vol. 14, no. 6, pp. 617–626, 2004. 2.1.1

[25] R. Oliveira, J. Clemente, A. Cunha, and M. Carrondo, "Adaptive dissolved oxygen control through the glycerol feeding in a recombinant pichia pastoris cultivation in conditions of oxygen transfer limitation," *Journal of biotechnology*, vol. 116, no. 1, pp. 35–50, 2005. 2.1.1

[26] I. D. Landau, R. Lozano, M. M'Saad, and A. Karimi, *Adaptive control: algorithms, analysis and applications*. Springer Science & Business Media, 2011. 2.1.1

[27] J. Lee, S. Y. Lee, S. Park, and A. P. Middelberg, "Control of fed-batch fermentations," *Biotechnology advances*, vol. 17, no. 1, pp. 29–48, 1999. 2.1.1

[28] R. Babuška and H. B. Verbruggen, "An overview of fuzzy modeling for control," *Control Engineering Practice*, vol. 4, no. 11, pp. 1593–1606, 1996. 2.1.1

[29] J.-I. Horiuchi and K. Hiraga, "Industrial application of fuzzy control to large-scale recombinant vitamin b2 production," *Journal of bioscience and bioengineering*, vol. 87, no. 3, pp. 365–371, 1999. 2.1.1

[30] X.-C. Zhang, A. Visala, A. Halme, and P. Linko, "Functional state modeling and fuzzy control of fed-batch aerobic baker's yeast process," *Journal of biotechnology*, vol. 37, no. 1, pp. 1–10, 1994. 2.1.1

[31] J. Glassey, G. Montague, A. Ward, and B. Kara, "Enhanced supervision of recombinant e. coli fermentation via artificial neural networks," *Process Biochemistry*, vol. 29, no. 5, pp. 387–398, 1994. 2.1.2

[32] J. H. Holland, "Genetic algorithms and adaptation," *Adaptive control of ill-defined systems*, pp. 317–333, 1984. 2.1.2

[33] L. Ferreira, M. De Souza Jr, and R. Folly, "Development of an alcohol fermentation control system based on biosensor measurements interpreted by neural networks," *Sensors and Actuators B: Chemical*, vol. 75, no. 3, pp. 166–171, 2001. 2.1.2

[34] L. Chen, S. K. Nguang, X. D. Chen, and X. M. Li, "Modelling and optimization of fed-batch fermentation processes using dynamic neural networks and genetic algorithms," *Biochemical Engineering Journal*, vol. 22, no. 1, pp. 51–61, 2004. 2.1.2

[35] J. Peng, F. Meng, and Y. Ai, "Time-dependent fermentation control strategies for enhancing synthesis of marine bacteriocin 1701 using artificial neural network and genetic algorithm," *Bioresource technology*, vol. 138, pp. 345–352, 2013. 2.1.2

[36] L. Cavagnari, L. Magni, and R. Scattolini, "Neural network implementation of nonlinear receding-horizon control," *Neural computing & applications*, vol. 8, pp. 86–92, 1999. 2.1.2

[37] S. Chen, K. Saulnier, N. Atanasov, D. D. Lee, V. Kumar, G. J. Pappas, and M. Morari, "Approximating explicit model predictive control using constrained neural networks," in *2018 Annual American control conference (ACC)*, pp. 1520–1527, IEEE, 2018. 2.1.2

[38] Y. Long, L. Liu, F. Shen, L. Shao, and X. Li, "Zero-shot learning using synthesised unseen visual data with diffusion regularisation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 40, no. 10, pp. 2498–2512, 2017. 2.1.3, 4.1, 4.2, 6.1

[39] S. Syafiie, F. Tadeo, and E. Martinez, "Model-free learning control of neutralization processes using reinforcement learning," *Engineering Applications of Artificial Intelligence*, vol. 20, no. 6, pp. 767–782, 2007. 2.1.3, 2.1.3, 2.1.3, 2.1.3

[40] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," 2011. 2.1.3, 3.1, 3.2

[41] R. Schoknecht and M. Riedmiller, "Learning to control at multiple time scales," in *Artificial Neural Networks and Neural Information Processing—ICANN/ICONIP 2003*, pp. 479–487, Springer, 2003. 2.1.3

[42] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018. 2.1.3

[43] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013. 2.1.3

[44] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015. 2.1.3, 4.2

[45] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double q-learning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, 2016. 2.1.3

[46] K. De Asis, J. Hernandez-Garcia, G. Holland, and R. Sutton, "Multi-step reinforcement learning: A unifying algorithm," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, 2018. 2.1.3

[47] M. Bertalmio, A. L. Bertozzi, and G. Sapiro, "Navier-stokes, fluid dynamics, and image and video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. I–I, IEEE, 2001. 2.2

[48] J.-B. Huang, S. B. Kang, N. Ahuja, and J. Kopf, "Temporally coherent completion of dynamic video," *TOG*, vol. 35, no. 6, pp. 1–11, 2016. 2.2, 5.1, 5.2, 5.3.1

[49] A. Newson, A. Almansa, M. Fradet, Y. Gousseau, and P. Pérez, "Video inpainting of complex scenes," *Siam journal on imaging sciences*, vol. 7, no. 4, pp. 1993–2019, 2014. 2.2, 2.2.1, 5.1, 5.2

[50] C. Wang, H. Huang, X. Han, and J. Wang, "Video inpainting by jointly learning temporal structure and spatial details," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, pp. 5232–5239, 2019. 2.2, 2.2.2, 5.1

[51] R. Xu, X. Li, B. Zhou, and C. C. Loy, "Deep flow-guided video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3723–3732, 2019. 2.2, 2.2.1, 2.2.2, 5.1, 5.2, 5.4.2, 5.1

[52] C. Barnes, E. Shechtman, A. Finkelstein, and D. B. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Trans. Graph.*, vol. 28, no. 3, p. 24, 2009. 2.2.1

[53] A. Criminisi, P. Pérez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Transactions on image processing*, vol. 13, no. 9, pp. 1200–1212, 2004. 2.2.1

[54] K. A. Patwardhan, G. Sapiro, and M. Bertalmio, "Video inpainting of occluding and occluded objects," in *IEEE International Conference on Image Processing 2005*, vol. 2, pp. II–69, IEEE, 2005. 2.2.1

[55] N. Xu, L. Yang, Y. Fan, D. Yue, Y. Liang, J. Yang, and T. Huang, "Youtube-vos: A large-scale video object segmentation benchmark," *arXiv preprint arXiv:1809.03327*, 2018. 2.2.1, 5.4.1

[56] D. Kim, S. Woo, J.-Y. Lee, and I. S. Kweon, "Deep video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5792–5801, 2019. 2.2.2, 5.2, 5.4.2, 5.1

[57] S. Lee, S. W. Oh, D. Won, and S. J. Kim, "Copy-and-paste networks for deep video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4413–4421, 2019. 2.2.2, 5.2

[58] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4471–4480, 2019. 2.2.2

[59] Y.-L. Chang, Z. Y. Liu, K.-Y. Lee, and W. Hsu, "Learnable gated temporal shift module for deep video inpainting," *arXiv preprint arXiv:1907.01131*, 2019. 2.2.2, 5.2, 5.4.2, 5.1

[60] H. Zhang, L. Mai, N. Xu, Z. Wang, J. Collomosse, and H. Jin, "An internal learning approach to video inpainting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2720–2729, 2019. 2.2.2

[61] Y.-T. Hu, H. Wang, N. Ballas, K. Grauman, and A. G. Schwing, "Proposal-based video completion," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVII 16*, pp. 38–54, Springer, 2020. 2.2.2

[62] X. Zou, L. Yang, D. Liu, and Y. J. Lee, "Progressive temporal feature alignment network for video inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16448–16457, 2021. 2.2.2

[63] C. Gao, A. Saraf, J.-B. Huang, and J. Kopf, "Flow-edge guided video completion," in *ECCV*, pp. 713–729, Springer, 2020. 2.2.2, 5.1, 5.1, 5.2, 5.4.2, 5.1

[64] T. H. Kim, M. S. Sajjadi, M. Hirsch, and B. Scholkopf, "Spatio-temporal transformer network for video restoration," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 106–122, 2018. 2.2.2

[65] J. Pan, H. Bai, and J. Tang, "Cascaded deep video deblurring using temporal sharpness prior," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3043–3051, 2020. 2.2.2

[66] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "Tdan: Temporally-deformable alignment network for video super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3360–3369, 2020. 2.2.2

[67] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *International Journal of Computer Vision*, vol. 127, pp. 1106–1125, 2019. 2.2.2

[68] Y. Matsushita, E. Ofek, W. Ge, X. Tang, and H.-Y. Shum, "Full-frame video stabilization with motion inpainting," *IEEE Transactions on pattern analysis and Machine Intelligence*, vol. 28, no. 7, pp. 1150–1163, 2006. 2.2.2

[69] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5791–5800, 2020. 2.2.2

[70] Y. Zeng, J. Fu, H. Chao, and B. Guo, "Learning pyramid-context encoder network for high-quality image inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1486–1494, 2019. 2.2.2

[71] A. Li, S. Zhao, X. Ma, M. Gong, J. Qi, R. Zhang, D. Tao, and R. Kotagiri, "Short-term and long-term context aggregation network for video inpainting," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pp. 728–743, Springer, 2020. 2.2.2

[72] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020. 2.2.2, 2.3.1

[73] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 10012–10022, 2021. 2.2.2

[74] J. Yang, C. Li, P. Zhang, X. Dai, B. Xiao, L. Yuan, and J. Gao, "Focal attention for long-range interactions in vision transformers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 30008–30022, 2021. 2.2.2

[75] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017. 2.2.2, 2.2.2, 2.2.2, 2.2.2, 2.3.1, 6.2, 6.4.2

[76] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, and R. Cucchiara, "From show to tell: a survey on deep learning-based image captioning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 1, pp. 539–559, 2022. 2.3.1

[77] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan, "Show and tell: A neural image caption generator," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3156–3164, 2015. 2.3.1, 2.3.2

[78] A. Karpathy and L. Fei-Fei, "Deep visual-semantic alignments for generating image descriptions," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3128–3137, 2015. 2.3.1, 6.1, 6.4.1

[79] J. Donahue, L. Anne Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell, "Long-term recurrent convolutional networks for visual recognition and description," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2625–2634, 2015. 2.3.1, 2.3.2, 6.2

[80] J. Mao, W. Xu, Y. Yang, J. Wang, Z. Huang, and A. Yuille, "Deep captioning with multimodal recurrent neural networks (m-rnn)," *arXiv preprint arXiv:1412.6632*, 2014. 2.3.1, 6.2

[81] X. Chen and C. Lawrence Zitnick, "Mind's eye: A recurrent visual representation for image caption generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2422–2431, 2015. 2.3.1

[82] H. Fang, S. Gupta, F. Iandola, R. K. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, *et al.*, "From captions to visual concepts and back," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1473–1482, 2015. 2.3.1

[83] X. Jia, E. Gavves, B. Fernando, and T. Tuytelaars, "Guiding the long-short term memory model for image caption generation," in *Proceedings of the IEEE international conference on computer vision*, pp. 2407–2415, 2015. 2.3.1

[84] S. J. Rennie, E. Marcheret, Y. Mroueh, J. Ross, and V. Goel, "Self-critical sequence training for image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7008–7024, 2017. 2.3.1, 6.2, 6.1, 6.4.3

[85] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016. 2.3.1, 6.4.2

[86] T. Yao, Y. Pan, Y. Li, Z. Qiu, and T. Mei, "Boosting image captioning with attributes," in *Proceedings of the IEEE international conference on computer vision*, pp. 4894–4902, 2017. 2.3.1

[87] Z. Gan, C. Gan, X. He, Y. Pu, K. Tran, J. Gao, L. Carin, and L. Deng, "Semantic compositional networks for visual captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5630–5639, 2017. 2.3.1

[88] B. Dai, D. Ye, and D. Lin, "Rethinking the form of latent states in image captioning," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 282–298, 2018. 2.3.1

[89] K. Wang, Q. Yin, W. Wang, S. Wu, and L. Wang, "A comprehensive survey on cross-modal retrieval," *arXiv preprint arXiv:1607.06215*, 2016. 2.3.1

[90] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6077–6086, 2018. 2.3.1, 2.3.2, 6.2, 6.4.2, 6.1, 6.4.3

[91] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *Advances in neural information processing systems*, vol. 28, 2015. 2.3.1, 6.4.2

[92] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma, *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *International journal of computer vision*, vol. 123, pp. 32–73, 2017. 2.3.1, 6.4.2

[93] L. Ke, W. Pei, R. Li, X. Shen, and Y.-W. Tai, "Reflective decoding network for image captioning," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8888–8897, 2019. 2.3.1

[94] Y. Qin, J. Du, Y. Zhang, and H. Lu, "Look back and predict forward in image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8367–8375, 2019. 2.3.1

[95] T. Yao, Y. Pan, Y. Li, and T. Mei, "Exploring visual relationship for image captioning," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 684–699, 2018. 2.3.1, 6.1, 6.2, 6.1, 6.4.3

[96] L. Guo, J. Liu, J. Tang, J. Li, W. Luo, and H. Lu, "Aligning linguistic words and visual semantic units for image captioning," in *Proceedings of the 27th ACM international conference on multimedia*, pp. 765–773, 2019. 2.3.1

[97] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016. 2.3.1

[98] X. Yang, K. Tang, H. Zhang, and J. Cai, "Auto-encoding scene graphs for image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10685–10694, 2019. 2.3.1, 6.2

[99] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, and R. Cucchiara, "From show to tell: a survey on deep learning-based image captioning," *IEEE transactions on pattern analysis and machine intelligence*, vol. 45, no. 1, pp. 539–559, 2022. 2.3.1

[100] X. Yang, H. Zhang, and J. Cai, "Learning to collocate neural modules for image captioning," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 4250–4260, 2019. 2.3.1

[101] G. Li, L. Zhu, P. Liu, and Y. Yang, "Entangled transformer for image captioning," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8928–8937, 2019. 2.3.1, 2.3.2

[102] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 801–818, 2018. 2.3.1

[103] Y. Ren, X. Yu, R. Zhang, T. H. Li, S. Liu, and G. Li, "Structureflow: Image inpainting via structure-aware appearance flow," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 181–190, 2019. 2.3.1

[104] L. Xu, Q. Yan, Y. Xia, and J. Jia, "Structure extraction from texture via relative total variation," *ACM transactions on graphics (TOG)*, vol. 31, no. 6, pp. 1–10, 2012. 2.3.1

[105] Q. Sun, L. Ma, S. J. Oh, L. Van Gool, B. Schiele, and M. Fritz, "Natural and effective obfuscation by head inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5050–5059, 2018. 2.3.1

[106] B. Rosenhahn and N. Pugeault, "Image captioning through image transformer," 2.3.1

[107] L. Huang, W. Wang, J. Chen, and X.-Y. Wei, "Attention on attention for image captioning," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4634–4643, 2019. 2.3.1, 2.3.2, 6.2, 6.1, 6.4.3

[108] Y. Pan, T. Yao, Y. Li, and T. Mei, "X-linear attention networks for image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10971–10980, 2020. 2.3.1, 2.3.2

[109] M. Cornia, M. Stefanini, L. Baraldi, and R. Cucchiara, "Meshed-memory transformer for image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10578–10587, 2020. 2.3.1, 2.3.2, 6.1, 6.2, 6.3.2, 6.1, 6.4.3

[110] Y. Luo, J. Ji, X. Sun, L. Cao, Y. Wu, F. Huang, C.-W. Lin, and R. Ji, "Dual-level collaborative transformer for image captioning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 2286–2293, 2021. 2.3.1, 2.3.2, 6.2

[111] W. Liu, S. Chen, L. Guo, X. Zhu, and J. Liu, "Cptr: Full transformer network for image captioning," *arXiv preprint arXiv:2101.10804*, 2021. 2.3.1

[112] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, *et al.*, "Learning transferable visual models from natural language supervision," in *International conference on machine learning*, pp. 8748–8763, PMLR, 2021. 2.3.1, 6.1, 6.2

[113] S. Shen, L. H. Li, H. Tan, M. Bansal, A. Rohrbach, K.-W. Chang, Z. Yao, and K. Keutzer, "How much can clip benefit vision-and-language tasks?," *arXiv preprint arXiv:2107.06383*, 2021. 2.3.1, 6.2, 6.1, 6.4.3

[114] R. Mokady, A. Hertz, and A. H. Bermano, "Clipcap: Clip prefix for image captioning," *arXiv preprint arXiv:2111.09734*, 2021. 2.3.1

[115] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "Vinvl: Revisiting visual representations in vision-language models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5579–5588, 2021. 2.3.1

[116] A. Graves and A. Graves, "Long short-term memory," *Supervised sequence labelling with recurrent neural networks*, pp. 37–45, 2012. 2.3.2

[117] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, pp. 2048–2057, PMLR, 2015. 2.3.2

[118] J. Lu, C. Xiong, D. Parikh, and R. Socher, "Knowing when to look: Adaptive attention via a visual sentinel for image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 375–383, 2017. 2.3.2

[119] X. Chen, L. Ma, W. Jiang, J. Yao, and W. Liu, "Regularizing rnns for caption generation by reconstructing the past with the present," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7995–8003, 2018. 2.3.2

[120] Y. Wang, Z. Lin, X. Shen, S. Cohen, and G. W. Cottrell, "Skeleton key: Image captioning by skeleton-attribute decomposition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7272–7281, 2017. 2.3.2

[121] J. Gu, J. Cai, G. Wang, and T. Chen, "Stack-captioning: Coarse-to-fine learning for image captioning," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 32, 2018. 2.3.2

[122] J. Lu, J. Yang, D. Batra, and D. Parikh, "Neural baby talk," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7219–7228, 2018. 2.3.2, 6.2

[123] S. Herdade, A. Kappeler, K. Boakye, and J. Soares, "Image captioning: Transforming objects into words," *Advances in neural information processing systems*, vol. 32, 2019. 2.3.2, 6.2, 6.1, 6.4.3

[124] L. Guo, J. Liu, X. Zhu, P. Yao, S. Lu, and H. Lu, "Normalized and geometry-aware self-attention network for image captioning," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10327–10336, 2020. 2.3.2

[125] Z. Wang, J. Yu, A. W. Yu, Z. Dai, Y. Tsvetkov, and Y. Cao, "Simvlm: Simple visual language model pretraining with weak supervision," *arXiv preprint arXiv:2108.10904*, 2021. 2.3.2

[126] J. Ji, Y. Luo, X. Sun, F. Chen, G. Luo, Y. Wu, Y. Gao, and R. Ji, "Improving image captioning by leveraging intra-and inter-layer global representation in transformer network," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 35, pp. 1655–1663, 2021. 2.3.2

[127] J. Zhang, "Modeling and optimal control of batch processes using recurrent neuro-fuzzy networks," *IEEE Transactions on Fuzzy Systems*, vol. 13, no. 4, pp. 417–427, 2005. 3.1, 3.4.2, 4.1, 4.3, 4.4.3

[128] Z. Xiong and J. Zhang, "Product quality trajectory tracking in batch processes using iterative learning control based on time-varying perturbation models," *Industrial & Engineering Chemistry Research*, vol. 42, no. 26, pp. 6802–6814, 2003. 3.1, 3.2, 3.4.1

[129] J. Zhang and R. Fisher, "Reliable multi-objective on-line re-optimisation control of a fed-batch fermentation process using bootstrap aggregated neural networks," in *Computer Science and Intelligent Controls (ISCSIC), 2017 International Symposium on*, pp. 49–56, IEEE, 2017. 3.1

[130] D. Li, L. Qian, Q. Jin, and T. Tan, "Reinforcement learning control with adaptive gain for a saccharomyces cerevisiae fermentation process," *Applied Soft Computing*, vol. 11, no. 8, pp. 4488–4495, 2011. 3.2

[131] M. J. Arauzo-Bravo, J. M. Cano-Izquierdo, E. Gomez-Sanchez, M. J. López-Nieto, Y. A. Dimitriadis, and J. López-Coronado, "Automatization of a penicillin production process with soft sensors and an adaptive controller based on neuro fuzzy systems," *Control Engineering Practice*, vol. 12, no. 9, pp. 1073–1090, 2004. 3.2

[132] Z. K. Nagy, "Model based control of a yeast fermentation bioreactor using optimally designed artificial neural networks," *Chemical engineering journal*, vol. 127, no. 1-3, pp. 95–109, 2007. 3.2

[133] Spielberg.Kumar, "Deep reinforcement learning approaches for process control," in *Advanced Control of Industrial Processes International Symposium on*, pp. 201–206, IEEE, 2017. 3.2, 4.1

[134] P. Zhang, J. Zhang, B. Hu, and Y. Long, "Optimization control of a fed-batch process using an improved reinforcement learning algorithm," in *IEEE Conference on Control Technology and Applications*, pp. 314–319, IEEE, 2019. 4.1, 4.1, 4.2, 4.3.2, 4.4.2, 4.3, 4.4.3

[135] P. Zhang, J. Zhang, Y. Long, and B. Hu, "An improved reinforcement learning control strategy for batch processes," in *2019 24th International Conference on Methods and Models in Automation and Robotics*, pp. 360–365, IEEE, 2019. 4.1

[136] P. Khalili, R. Vatankhah, and S. Taghvaei, "Optimal sliding mode control of drug delivery in cancerous tumour chemotherapy considering the obesity effects," *IET Systems Biology*, vol. 12, no. 4, pp. 185–189, 2018. 4.1

[137] J. Carius, R. Ranftl, V. Koltun, and M. Hutter, "Trajectory optimization with implicit hard contacts," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3316–3323, 2018. 4.1

[138] Q. Wei, X.-y. Wang, and X.-P. Hu, "Optimal control for permanent magnet synchronous motor," *Journal of Vibration and Control*, vol. 20, no. 8, pp. 1176–1184, 2014. 4.1

[139] B. Hu, Y. Gao, Y. Guan, Y. Long, N. Lane, and T. Ploetz, "Robust cross-view gait identification with evidence: A discriminant gait GAN (DIGGAN) approach on 10000 people," *arXiv preprint arXiv:1811.10493*, 2018. 4.1

[140] M. Jaderberg, W. M. Czarnecki, I. Dunning, L. Marris, G. Lever, A. G. Castaneda, C. Beattie, N. C. Rabinowitz, A. S. Morcos, A. Ruderman, *et al.*, "Human-level performance in 3D multiplayer games with population-based reinforcement learning," *Science*, vol. 364, no. 6443, pp. 859–865, 2019. 4.1, 4.2

[141] Y. Gao, Y. Long, Y. Guan, A. Basu, J. Baggaley, and T. Ploetz, "Towards reliable, automated general movement assessment for perinatal stroke screening in infants using wearable accelerometers," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 3, no. 1, pp. 1–22, 2019. 4.1

[142] I. Atasoy, M. Yuceer, and R. Berber, "Optimisation of operating conditions in fed-batch baker's yeast fermentation," *Chemical and Process Engineering*, vol. 34, no. 1, pp. 175–186, 2013. 4.2

[143] W. Tochampa, S. Sirisansaneeyakul, W. Vanichsriratana, P. Srinophakun, H. H. Bakker, S. Wannawilai, and Y. Chisti, "Optimal control of feeding in fed-batch production of xylitol," *Industrial & Engineering Chemistry Research*, vol. 54, no. 7, pp. 1992–2000, 2015. 4.2

[144] S. Goldrick, A. Ştefan, D. Lovett, G. Montague, and B. Lennox, "The development of an industrial-scale fed-batch fermentation simulation," *Journal of Biotechnology*, vol. 193, pp. 70–82, 2015. 4.2

[145] J. R. Banga, E. Balsa-Canto, C. G. Moles, and A. A. Alonso, "Dynamic optimization of bioprocesses: Efficient and robust numerical strategies," *Journal of Biotechnology*, vol. 117, no. 4, pp. 407–419, 2005. 4.2

[146] C. Tai, D. R. Keshwani, D. S. Voltan, P. S. Kuhar, and A. J. Engel, "Optimal control strategy for fed-batch enzymatic hydrolysis of lignocellulosic biomass based on epidemic modeling," *Biotechnology and Bioengineering*, vol. 112, no. 7, pp. 1376–1382, 2015. 4.2

[147] E. C. Martínez, M. D. Cristaldi, and R. J. Grau, "Dynamic optimization of bioreactors using probabilistic tendency models and Bayesian active learning," *Computers & Chemical Engineering*, vol. 49, pp. 37–49, 2013. 4.2

[148] S. Ochoa, "A new approach for finding smooth optimal feeding profiles in fed-batch fermentations," *Biochemical Engineering Journal*, vol. 105, pp. 177–188, 2016. 4.2

[149] S. Craven, J. Whelan, and B. Glennon, "Glucose concentration control of a fed-batch mammalian cell bioprocess using a nonlinear model predictive controller," *Journal of Process Control*, vol. 24, no. 4, pp. 344–357, 2014. 4.2

[150] E. A. del Rio-Chanona, D. Zhang, and V. S. Vassiliadis, "Model-based real-time optimisation of a fed-batch cyanobacterial hydrogen production process using economic model predictive control strategy," *Chemical Engineering Science*, vol. 142, pp. 289–298, 2016. 4.2

[151] L. Dewasme, S. Fernandes, Z. Amribt, L. Santos, P. Bogaerts, and A. V. Wouwer, "State estimation and predictive control of fed-batch cultures of hybridoma cells," *Journal of Process Control*, vol. 30, pp. 50–57, 2015. 4.2

[152] H. Duan, S. Wang, and Y. Guan, "Sofa-net: Second-order and first-order attention network for crowd counting," *arXiv preprint arXiv:2008.03723*, 2020. 4.2

[153] Z. Wang, X. Li, H. Duan, and X. Zhang, "A self-supervised residual feature learning model for multifocus image fusion," *IEEE Transactions on Image Processing*, vol. 31, pp. 4527–4542, 2022. 4.2

[154] M. Ryu, Y. Chow, R. Anderson, C. Tjandraatmadja, and C. Boutilier, "CAQL: Continuous action Q-learning," *arXiv preprint arXiv:1909.12397*, 2019. 4.2

[155] K. Hausman, J. T. Springenberg, Z. Wang, N. Heess, and M. Riedmiller, "Learning an embedding space for transferable robot skills," in *International Conference on Learning Representations*, 2018. 4.2

[156] C. Florensa, Y. Duan, and P. Abbeel, "Stochastic neural networks for hierarchical reinforcement learning," *arXiv preprint arXiv:1704.03012*, 2017. 4.2

[157] M. Kearns and S. Singh, "Near-optimal reinforcement learning in polynomial time," *Machine Learning*, vol. 49, no. 2-3, pp. 209–232, 2002. 4.2

[158] A. Gupta, R. Mendonca, Y. Liu, P. Abbeel, and S. Levine, "Meta-reinforcement learning of structured exploration strategies," in *Advances in Neural Information Processing Systems*, pp. 5302–5311, 2018. 4.2

[159] F. Garcia and P. S. Thomas, "A meta-MDP approach to exploration for lifelong reinforcement learning," in *Advances in Neural Information Processing Systems*, pp. 5691–5700, 2019. 4.2

[160] A. G. Van Hasselt, Hado and D. Silver, "Deep reinforcement learning with double Q-learning in Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence," 2016. 4.2

[161] Z. Wang, T. Schaul, M. Hessel, H. Hasselt, M. Lanctot, and N. Freitas, "Dueling network architectures for deep reinforcement learning," in *International Conference on Machine Learning*, pp. 1995–2003, 2016. 4.2

[162] F. S. He, Y. Liu, A. G. Schwing, and J. Peng, "Learning to play in a day: Faster deep reinforcement learning by optimality tightening," *arXiv preprint arXiv:1611.01606*, 2016. 4.2

[163] A. Harutyunyan, M. G. Bellemare, T. Stepleton, and R. Munos, "Q(λ) with off-policy corrections," in *International Conference on Algorithmic Learning Theory*, pp. 305–320, Springer, 2016. 4.2

[164] R. Munos, T. Stepleton, A. Harutyunyan, and M. Bellemare, "Safe and efficient off-policy reinforcement learning," in *Advances in Neural Information Processing Systems*, pp. 1054–1062, 2016. 4.2

[165] M. Fortunato, M. G. Azar, B. Piot, J. Menick, I. Osband, A. Graves, V. Mnih, R. Munos, D. Hassabis, O. Pietquin, *et al.*, "Noisy networks for exploration," *arXiv preprint arXiv:1706.10295*, 2017. 4.2

[166] M. G. Bellemare, W. Dabney, and R. Munos, "A distributional perspective on reinforcement learning," *arXiv preprint arXiv:1707.06887*, 2017. 4.2

[167] A. Pritzel, B. Uria, S. Srinivasan, A. Puigdomenech, O. Vinyals, D. Hassabis, D. Wierstra, and C. Blundell, "Neural episodic control," *arXiv preprint arXiv:1703.01988*, 2017. 4.2

[168] Z. Lin, T. Zhao, G. Yang, and L. Zhang, "Episodic memory deep Q-networks," *arXiv preprint arXiv:1805.07603*, 2018. 4.2

[169] M. L. Littman, "Value-function reinforcement learning in Markov games," *Cognitive Systems Research*, vol. 2, no. 1, pp. 55–66, 2001. 4.2

[170] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *Journal of machine learning research*, vol. 4, no. Nov, pp. 1039–1069, 2003. 4.2

[171] M. Lauer and M. Riedmiller, "An algorithm for distributed reinforcement learning in cooperative multi-agent systems," in *In Proceedings of the Seventeenth International Conference on Machine Learning*, Citeseer, 2000. 4.2

[172] G. Arslan and S. Yüksel, "Decentralized Q-learning for stochastic teams and games," *IEEE Transactions on Automatic Control*, vol. 62, no. 4, pp. 1545–1558, 2016. 4.2

[173] N. Bard, J. N. Foerster, S. Chandar, N. Burch, M. Lanctot, H. F. Song, E. Parisotto, V. Dumoulin, S. Moitra, E. Hughes, *et al.*, "The Hanabi challenge: A new frontier for AI research," *Artificial Intelligence*, vol. 280, p. 103216, 2020. 4.2

[174] J. Foerster, F. Song, E. Hughes, N. Burch, I. Dunning, S. Whiteson, M. Botvinick, and M. Bowling, "Bayesian action decoder for deep multi-agent reinforcement learning," in *International Conference on Machine Learning*, pp. 1942–1951, 2019. 4.2

[175] D. Lee, H. Yoon, and N. Hovakimyan, "Primal-dual algorithm for distributed reinforcement learning: distributed GTD," in *IEEE Conference on Decision and Control*, pp. 1967–1972, IEEE, 2018. 4.2

[176] J. Foerster, I. A. Assael, N. De Freitas, and S. Whiteson, "Learning to communicate with deep multi-agent reinforcement learning," in *Advances in Neural Information Processing Systems*, pp. 2137–2145, 2016. 4.2

[177] J. Foerster, N. Nardelli, G. Farquhar, T. Afouras, P. H. Torr, P. Kohli, and S. Whiteson, "Stabilising experience replay for deep multi-agent reinforcement learning," *arXiv preprint arXiv:1702.08887*, 2017. 4.2

[178] J. K. Gupta, M. Egorov, and M. Kochenderfer, "Cooperative multi-agent control using deep reinforcement learning," in *International Conference on Autonomous Agents and Multiagent Systems*, pp. 66–83, Springer, 2017. 4.2

[179] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor-critic for mixed cooperative-competitive environments," in *Advances in Neural Information Processing Systems*, pp. 6379–6390, 2017. 4.2

[180] S. Omidshafiei, J. Pazis, C. Amato, J. P. How, and J. Vian, "Deep decentralized multi-task multi-agent reinforcement learning under partial observability," *arXiv preprint arXiv:1703.06182*, 2017. 4.2

[181] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Transactions on Cybernetics*, 2020. 4.2

[182] E. Wei, D. Wicke, D. Freelan, and S. Luke, "Multiagent soft Q-learning," *arXiv preprint arXiv:1804.09817*, 2018. 4.2

[183] E. Wei and S. Luke, "Lenient learning in independent-learner stochastic cooperative games," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 2914–2955, 2016. 4.2

[184] S. Sukhbaatar and R. Fergus, "Learning multiagent communication with backpropagation," in *Advances in Neural Information Processing Systems*, pp. 2244–2252, 2016. 4.2

[185] J. Jiang and Z. Lu, "Learning attentional communication for multi-agent cooperation," in *Advances in Neural Information Processing Systems*, pp. 7254–7264, 2018. 4.2

[186] D. Kim, S. Moon, D. Hostallero, W. J. Kang, T. Lee, K. Son, and Y. Yi, "Learning to schedule communication in multi-agent reinforcement learning," *arXiv preprint arXiv:1902.01554*, 2019. 4.2

[187] K. Son, D. Kim, W. J. Kang, D. E. Hostallero, and Y. Yi, "QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning," *arXiv preprint arXiv:1905.05408*, 2019. 4.2

[188] H.-T. Wai, Z. Yang, Z. Wang, and M. Hong, "Multi-agent reinforcement learning via double averaging primal-dual optimization," in *Advances in Neural Information Processing Systems*, pp. 9649–9660, 2018. 4.2

[189] C. Qu, S. Mannor, H. Xu, Y. Qi, L. Song, and J. Xiong, "Value propagation for decentralized networked deep multi-agent reinforcement learning," in *Advances in Neural Information Processing Systems*, pp. 1184–1193, 2019. 4.2

[190] X. Liao, W. Li, Q. Xu, X. Wang, B. Jin, X. Zhang, Y. Wang, and Y. Zhang, "Iteratively-refined interactive 3D medical image segmentation with multi-agent reinforcement learning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9394–9402, 2020. 4.2

[191] M. R. Minar and J. Naher, "Recent advances in deep learning: An overview," *arXiv preprint arXiv:1807.08169*, 2018. 4.3.1

[192] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "Edvr: Video restoration with enhanced deformable convolutional networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 0–0, 2019. 5.1

[193] Y. Zeng, J. Fu, and H. Chao, "Learning joint spatial-temporal transformations for video inpainting," in *ECCV*, pp. 528–543, Springer, 2020. 5.1, 5.1, 5.2, 5.4.1, 5.4.2, 5.1, 5.4.2

[194] O. Fried, A. Tewari, M. Zollhöfer, A. Finkelstein, E. Shechtman, D. B. Goldman, K. Genova, Z. Jin, C. Theobalt, and M. Agrawala, "Text-based editing of talking-head video," *TOG*, vol. 38, no. 4, pp. 1–14, 2019. 5.1

[195] A. Lim, B. Ramesh, Y. Yang, C. Xiang, Z. Gao, and F. Lin, "Real-time optical flow-based video stabilization for unmanned aerial vehicles," *Journal of Real-Time Image Processing*, vol. 16, no. 6, pp. 1975–1985, 2019. 5.1

[196] H. Ouyang, T. Wang, and Q. Chen, "Internal video inpainting by implicit long-range propagation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14579–14588, 2021. 5.1

[197] Y. Ding, C. Wang, H. Huang, J. Liu, J. Wang, and L. Wang, "Frame-recurrent video inpainting by robust optical flow inference," *arXiv preprint arXiv:1905.02882*, 2019. 5.1

[198] M. Bertalmio, L. Vese, G. Sapiro, and S. Osher, "Simultaneous structure and texture image inpainting," *TIP*, vol. 12, no. 8, pp. 882–889, 2003. 5.2

[199] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen, "Image melding: Combining inconsistent images using patch-based synthesis," *TOG*, vol. 31, no. 4, pp. 1–10, 2012. 5.2

[200] A. A. Efros and W. T. Freeman, "Image quilting for texture synthesis and transfer," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pp. 341–346, 2001. 5.2

[201] D. Simakov, Y. Caspi, E. Shechtman, and M. Irani, "Summarizing visual data using bidirectional similarity," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2008. 5.2

[202] T. K. Shih, N. C. Tang, and J.-N. Hwang, "Exemplar-based video inpainting without ghost shadow artifacts by maintaining temporal continuity," *T-CSVT*, vol. 19, no. 3, pp. 347–360, 2009. 5.2

[203] M. Strobel, J. Diebold, and D. Cremers, "Flow and color inpainting for video completion," in *German Conference on Pattern Recognition*, pp. 293–304, Springer, 2014. 5.2

[204] T. Kim, Y. Cho, D. Kim, M. Chang, and Y.-J. Kim, "Tooth segmentation of 3d scan data using generative adversarial networks," *Applied Sciences*, vol. 10, no. 2, p. 490, 2020. 5.2

[205] J. S. Ren, L. Xu, Q. Yan, and W. Sun, "Shepard convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 28, pp. 901–909, 2015. 5.2

[206] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2536–2544, 2016. 5.2

[207] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Globally and locally consistent image completion," *TOG*, vol. 36, no. 4. 5.2

[208] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Generative image inpainting with contextual attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5505–5514, 2018. 5.2

[209] S. W. Oh, S. Lee, J.-Y. Lee, and S. J. Kim, "Onion-peel networks for deep video completion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4403–4412, 2019. 5.2

[210] A. Bokov and D. Vatolin, "100+ times faster video completion by optical-flow-guided variational refinement," in *ICIP*, pp. 2122–2126, IEEE, 2018. 5.2, 5.3.1

[211] M. Okabe, K. Noda, Y. Dobashi, and K. Anjyo, "Interactive video completion," *IEEE computer graphics and applications*, vol. 40, no. 1, pp. 127–139, 2019. 5.2, 5.3.1

[212] E. Ilg, N. Mayer, T. Saikia, M. Keuper, A. Dosovitskiy, and T. Brox, "Flownet 2.0: Evolution of optical flow estimation with deep networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2462–2470, 2017. 5.2

[213] Y. Bai, J. Wang, Y. Long, B. Hu, Y. Song, M. Pagnucco, and Y. Guan, "Discriminative latent semantic graph for video captioning," in *ACM MM*, pp. 3556–3564, 2021. 5.3.1

[214] A. Nagrani, S. Yang, A. Arnab, A. Jansen, C. Schmid, and C. Sun, "Attention bottlenecks for multimodal fusion," *Advances in Neural Information Processing Systems*, vol. 34, pp. 14200–14213, 2021. 5.3.1

[215] A. Arnab, M. Dehghani, G. Heigold, C. Sun, M. Lučić, and C. Schmid, "Vivit: A video vision transformer," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6836–6846, 2021. 5.3.1

[216] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *TIP*, vol. 13, no. 4, pp. 600–612, 2004. 5.4.1

[217] A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, and I. Sutskever, "Zero-shot text-to-image generation," in *International Conference on Machine Learning*, pp. 8821–8831, PMLR, 2021. 6.1

[218] Y. Huang, Y. Long, and L. Wang, "Few-shot image and sentence matching via gated visual-semantic embedding," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 8489–8496, 2019. 6.1

[219] B. Klein, G. Lev, G. Sadeh, and L. Wolf, "Fisher vectors derived from hybrid gaussian-laplacian mixture models for image annotation," *arXiv preprint arXiv:1411.7399*, 2014. 6.2

[220] I. Schwartz, S. Yu, T. Hazan, and A. G. Schwing, "Factor graph attention," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2039–2048, 2019. 6.2

[221] I. Schwartz, A. G. Schwing, and T. Hazan, "A simple baseline for audio-visual scene-aware dialog," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12548–12558, 2019. 6.2

[222] M. Ding, Z. Yang, W. Hong, W. Zheng, C. Zhou, D. Yin, J. Lin, X. Zou, Z. Shao, H. Yang, *et al.*, "Cogview: Mastering text-to-image generation via transformers," *Advances in Neural Information Processing Systems*, vol. 34, pp. 19822–19835, 2021. 6.2

[223] X. Li, X. Yin, C. Li, P. Zhang, X. Hu, L. Zhang, L. Wang, H. Hu, L. Dong, F. Wei, *et al.*, "Oscar: Object-semantics aligned pre-training for vision-language tasks," in *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXX 16*, pp. 121–137, Springer, 2020. 6.2

[224] P. Zhang, X. Li, X. Hu, J. Yang, L. Zhang, L. Wang, Y. Choi, and J. Gao, "Vinvl: Making visual representations matter in vision-language models," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021. 6.2

[225] L. A. Hendricks, S. Venugopalan, M. Rohrbach, R. Mooney, K. Saenko, and T. Darrell, "Deep compositional captioning: Describing novel object categories without paired training data," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1–10, 2016. 6.2

[226] S. Venugopalan, L. Anne Hendricks, M. Rohrbach, R. Mooney, T. Darrell, and K. Saenko, "Captioning images with diverse objects," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5753–5761, 2017. 6.2

[227] Q. Feng, Y. Wu, H. Fan, C. Yan, M. Xu, and Y. Yang, "Cascaded revision network for novel object captioning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 10, pp. 3413–3421, 2020. 6.2

[228] Y. Feng, L. Ma, W. Liu, and J. Luo, "Unsupervised image captioning," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4125–4134, 2019. 6.2

[229] I. Laina, C. Rupprecht, and N. Navab, "Towards unsupervised image captioning with shared multimodal embeddings," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7414–7424, 2019. 6.2

[230] H. Chefer, S. Benaim, R. Paiss, and L. Wolf, "Image-based clip-guided essence transfer," in *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XIII*, pp. 695–711, Springer, 2022. 6.2

[231] A. Sanghi, H. Chu, J. G. Lambourne, Y. Wang, C.-Y. Cheng, M. Fumero, and K. R. Malekshan, "Clip-forge: Towards zero-shot text-to-shape generation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18603–18613, 2022. 6.2

[232] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002. 6.4.2

[233] S. Banerjee and A. Lavie, "Meteor: An automatic metric for mt evaluation with improved correlation with human judgments," in *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pp. 65–72, 2005. 6.4.2

[234] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, pp. 74–81, 2004. 6.4.2

[235] R. Vedantam, C. Lawrence Zitnick, and D. Parikh, "Cider: Consensus-based image description evaluation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4566–4575, 2015. 6.4.2

[236] W. Jiang, L. Ma, Y.-G. Jiang, W. Liu, and T. Zhang, "Recurrent fusion network for image captioning," in *Proceedings of the European conference on computer vision (ECCV)*, pp. 499–515, 2018. 6.1, 6.4.3