

**From Machine to Human Visual Motion
Processing: Exploring and Bridging the Gap
by Modelling**

Zitang Sun

June 2024

Declaration

I, Zitang Sun, declare that this thesis, entitled “From Machine to Human Visual Motion Processing: Exploring and Bridging the Gap by Modelling” is original and my own work. I confirm that:

- This work was done solely while I was a candidate for the research degree at the Graduate School of Informatics, Kyoto University.
- No part of this work has previously been submitted for a degree at this or any other university.
- References to the works of others have been clearly attributed. Quotations from the work of others have been clearly indicated and attributed to them.
- In cases where others have contributed to parts of this work, such contributions have been clearly acknowledged and distinguished from my own work.

Zitang Sun
August 2024

Abstract

Visual motion estimation includes the ability to infer the speed and direction of objects/elements in a scene based on the dynamic inputs from a visual sensor like a camera or eye. Such a function is crucial for humans and animals to navigate and interact with the dynamic world, and it is equally vital in the engineering field for machine/AI -agents to process downstream visual tasks, like object tracking, video interpolation, state prediction, etc. As such, visual motion estimation has garnered attention in both human science and engineering fields, such as human visual motion perception in vision science (VS) societies and optical/scene flow estimation in computer vision (CV).

Although many related studies are carried out in both VS and CV fields, there is a noticeable disparity between the two societies. Studies in VS primarily focus on the scientific understanding of the mechanisms behind human visual motion processing systems. Although many computational models have been proposed, few VS models are capable of handling complex scenarios in natural environments, such as estimating motion flow in the challenging conditions encountered in the field of CV. On the other hand, motion estimation has made significant progress in the field of CV, as demonstrated by the recent Raft and FlowFormer, which are based on deep learning techniques and the powerful representation capabilities of deep neural networks (DNNs). However, these models, which focus only on maximizing local image correspondences between frame pairs, often overlook the alignment with human perception. They fail to account for several fundamental phenomena observed in human vision, such as systematic or adaptive deviations of human perceived motion from local ground truth (GT), reverse-phi phenomena, and second-order motion. In addition, the recent CV models exhibit instability when dealing with non-textured stimuli commonly used in VS.

This thesis examines the mechanisms of visual motion estimation at the intersection between the CV and VS. The final goal is minimizing the Gap between CV and human vision systems by constructing a generalized motion processing model that merges the intelligence from deep learning and vision science, seeking both the functional and neurophysiological similarity to human vision systems and engineering applicability to complex natural scenarios.

The initial Chapter of this thesis provides a general introduction to human vision motion perception and optical flow estimation, emphasizing the significance of building a human-aligned motion processing model. This Chapter also explores the current situation and context of research around this area, highlighting the progress made and identifying persistent gaps between the two fields. (Chapter 1)

Subsequently, the thesis introduces the preliminaries associated with visual motion perception and optical flow estimation tasks. Specifically, it describes the current methodology of supervised and unsupervised learning for optical flow and identifies potential problems. Concurrently, it introduces basic computational methods for motion perception in both CV fields—such as the Lucas-Kanade method, DNNs-based methods like FlowNet, RAFT, and FlowFormer—and in VS fields, which include models based on Motion From Fourier Component (MFFC) principles and higher-order motion perception such as second-order motion perception. These discussions lay a solid foundation for subsequent sections (Chapter 2)

Chapter 3 reports a CV study that explores human-motivated unsupervised learning for motion estimation. This approach has the potential to imitate the learning behavior of humans and animals to achieve accurate motion estimation ability without explicit ground truth. Additionally, unsupervised learning has emerged as a viable alternative to supervised methods in model training, particularly because ground truth data is usually unavailable in natural scenarios. However, challenges arise with current unsupervised learning methods in the CV field, such as instability due to occlusion, motion blur, or varying image content over time, which disrupts pixel tracking and correspondence between frames. Recognizing that the human vision system behaves more like a temporal recurrent model to estimate motion in a continuous time-space, we propose a human-motivated unsupervised learning method based on multi-frame environment and temporal dynamic models, coping with the challenge scenes by incorporating motion temporal history and temporal continuity. Experiments on several computer vision optical flow benchmarks demonstrate the effectiveness of the proposed method compared to previous methods.

The human-motivated strategy demonstrates effectiveness in addressing problems in the CV field. However, such CV-oriented methods still have a gap when compared to human motion perception. In Chapter 4, we show the difference between CV models and human perception in motion perception tasks. We incorporate several representative CV models, including supervised and unsupervised methods, and compare them with a human perceptual response based on the in-silico psychophysical method. We show that most of the DNN-based models do not generalize to simple experimental stimuli, and DNN-based models cannot account for perceptual motion reversals perceived by humans with contrast-reversed patterns and a missing fundamental grating. In addition, none of the tested models can explain occlusion-aware motion

integration and second-order motion. Generally, there is a gap between the current CV motion models and the human visual motion perception system.

In Chapter 5, we propose a human-aligned model of human motion perception by bridging the gap between biological and CV models. Specifically, we introduce a novel two-stage approach that combines trainable motion energy sensing with a recurrent self-attention network for adaptive motion integration and segregation. This model architecture aims to capture the computations in V1-MT, the core structure for motion perception in the biological visual system, while providing the ability to derive informative motion flow for a wide range of stimuli, including complex natural scenes. Also, we show that the model’s unit responses are similar to mammalian neural recordings regarding motion pooling and speed tuning based on silico neurophysiology and demonstrate a good ability to predict human responses better than the ground truth, whereas the state-of-the-art CV models show the opposite. In addition to supervised training, we enable the model to learn motion estimation from unlabeled natural videos by incorporating the multi-frame unsupervised approach proposed in Chapter 3.

Our initial motion model is capable of replicating various motion phenomena of human perception. However, it is limited to first-order motion perception. Many animals, including flies, cats, and humans, have the ability to visually recognize image motion not only from shifts of spatial patterns defined by luminance modulations (first-order motion) but also from those defined by high-level image features such as temporal modulations and contrast modulations (second-order motion). In the past, second-order motion perception has been extensively studied using carefully designed artificial stimuli (e.g., drift-balanced motion) to control first-order motion components, but why and how the visual system has acquired this perceptual ability in natural environments remains poorly understood. We hypothesized that the biological system might naturally learn second-order motion perception for the purpose of estimating correct physical object motion amidst internal optical fluctuations, for example, those produced by highlights of glossy materials and refractions of transparent materials. As a proof of concept, we developed a DNN-based model to process both first- and second-order motions in natural scenes. We created initial second-order benchmarks and trained the model using specific optical flow datasets with varying material properties. The findings from multiple experiments suggest that second-order motion perception is significantly related to the material properties of training environments. The second-order motion system might have evolved, at least partially, to help in the robust estimation of object motion while countering optical turbulence under natural environments. (Chapter 6).

To conclude, this dissertation starts with machine vision and proposes a human-motivated CV study, improving the performance of unsupervised learning of motion estimation in the engineering field. Further, we shift toward the gap between human vision and current machine models

and construct a human-aligned model to bridge this gap. Specifically, we propose an image-computable model that simulates human motion perception and is capable of accommodating both first-order and second-order motions. This model exhibits neurophysiological and perceptual similarities to previous observations. It bridges computer vision models and human vision science by combining past visual science findings with advanced deep learning technologies. This is the first framework to effectively generalize both first- and second-order motion phenomena, accounting for a wide range of human visual motion perception, including multiple motion illusions, motion integration, second-order motion, and naturalistic scene perception. We will show that the development of a human-like machine vision model not only focuses on achieving technical accuracy but also enhances scientific understanding of the human visual system, inspires robust motion perception models in the engineering field, and improves human-machine interactions.

Acknowledgments

As I reflect upon my third-year journey into vision science research, I am profoundly grateful. Three years ago, amidst the turmoil of the COVID-19 pandemic, I hastily purchased a plane ticket and arrived in Japan just before my visa expired, embarking on my PhD journey. At that time, I was mired in confusion and self-doubt, questioning the value and significance of my work through the lens of an application-oriented and engineering mindset. However, my relentless pursuit of knowledge and the accompanying anxiety served as guiding lights, helping me emerge from the depths of isolation and ultimately complete my PhD research, despite the arduous process.

First and foremost, I extend my heartfelt gratitude to my supervisor, Shin'ya Nishida. When I was lost in this field, he reached out and pulled me up. He helped me understand the difference between engineering and science and taught me how to think scientifically. During times when I doubted my research, he consistently encouraged me to keep trying. His unwavering support and ample resources during my paper submissions to journals and international conferences enabled me to navigate and persevere through every challenge. Without his guidance, I would never have ventured into such an area of research.

Next, I wish to express my deepest appreciation to Professor Yukiyasu Kamitani. As my fellowship mentor, he generously shared his insights and guidance, regularly checking in on my research progress. He also encouraged me to engage in discussions with students in his lab. I am particularly grateful for my interactions with Ken Shirakawa and Fan Cheng.

I also wish to acknowledge other laboratory staff members, Hiroaki Mizuhara, Kiyofumi Miyoshi, and Kayo Yoshimura, for their valuable advice and guidance on my life and research. I extend my thanks to my fellow students, Yen-ju Chen, Yuan Li, Ziwei He, and Hanyang Jiang, for their cherished academic collaborations and camaraderie in both academic and personal spheres.

I am grateful to the department's teachers, including Hiroaki Mizuhara, Kiyofumi Miyoshi, Ko Nishino, Yukiyasu Kamitani, and others. Their valuable courses helped me understand fundamental statistical analysis methods and experimental design protocols in psychophysics and bioinformatics, as well as optical flow estimation and depth perception in computer vision, laying a solid foundation for my research.

I am grateful to my academic collaborators, Yen-ju Chen, Yun-hao Yang, Zhengbo Luo,

Fuli Cao, and others, with whom I have worked on various projects, enriching my experience in numerous ways; I extend my heartfelt appreciation to my internship mentor, Taiki Fukiage, for his tremendous support during my internship in NTT; I am grateful to my family, especially my parents, for their unwavering care and concern; Lastly, I would like to thank Han Xu for being by my side throughout my PhD journey. I am truly honored to have found both something to love, the realm of research, and someone to cherish during this period of my life.

I must acknowledge that I am neither intelligent enough to establish any profound theory nor diligent enough to provide significant empirical results. My contributions to the community may be limited. However, I am still glad to have pondered these thoughts and recorded them in my thesis. This is what I cherish most, as the life of a PhD dissertation is long, perhaps continuing to radiate warmth somewhere even after my passing. After such a journey, I accept the fact that I am an ordinary person—not in a helpless, negative way, but rather with pride. I live with it, learn about it, and am proud of it.

Best wishes to all shining mortals.

Contents

Declaration	i
Abstract	iii
Acknowledgments	vii
1 Introduction	1
1.1 Object of This Study	2
1.2 Modeling Motion Perception in Vision Science	3
1.3 Motion Estimation in Computer Vision	4
1.4 Current Problems and the Organization	6
2 Preliminary Background	11
2.1 Description of Motion	12
2.2 Early Findings in Motion Perception	12
2.3 Modelling Visual Motion Processing	14
2.3.1 Local Motion Detector	14
2.3.2 Second-order motion	20
2.3.3 Motion Integration	22
2.4 Motion Estimation based on DNNs	24
2.4.1 Common Optical Flow Datasets	25
2.4.2 Model Development	26
2.4.3 Unsupervised Learning Methods	31
3 Elaboration of Machine Vision: Human-motivated Unsupervised Learning of Motion Estimation	35
3.1 Problem Setting and Motivation	36
3.1.1 Problem I: Motion is a Dynamic Process	37
3.1.2 Problem II: Motion is not only Correspondence Matching	37
3.1.3 Proposal I: Temporal Dynamic Modelling & Scene Augmentation	37
3.1.4 Proposal II: A Decoupled Spatial-Temporal Adaptive Fusion Network	39

3.2	Discussion with Related Works	40
3.3	Temporal Dynamic Modelling & Scene Augmentation	41
3.3.1	Network structure	42
3.3.2	Occlusion-aware Temporal Smoothness Regularization.	44
3.3.3	Training in the Video Sequence	45
3.3.4	Dynamic Scene Augmentation	46
3.3.5	Self-supervised Distillation	51
3.4	Experiments I	52
3.4.1	Implementation Details	52
3.4.2	Comparison of the Sintel & KITTI benchmarks	55
3.4.3	Ablation Study	56
3.5	Decoupled Spatiotemporal Fusion Network	60
3.5.1	Global Correspondence Matching	61
3.5.2	Spatial-Temporal Adaptive Fusion	64
3.5.3	Decoupled Training	67
3.5.4	Efficient Training Step	69
3.6	Experiment II	71
3.6.1	Implementation Details	71
3.6.2	Ablation Study	71
3.6.3	Comparison with State-of-the-art	73
3.6.4	Augumentation on Natural Image	75
3.6.5	Comparison with Prototypes	75
3.7	Conclusion	75
4	Limitations of Current Machine Vision Models to Predict Human Motion Perception	77
4.1	Introduction	78
4.2	Test on Simple Stimuli	79
4.3	Mixed Training Strategy	79
4.4	Motion Illusions	82
4.5	Motion Integration	86
4.5.1	Motion Integration on Plaid	86
4.5.2	Motion Integration with Frame and Occlusion Constraints	86
4.5.3	Motion Integration with Recurrent Process	87
4.6	Conclusion and Discussion	88
5	Human-aligned Visual Motion Processing Model: First-order motion	91
5.1	Introduction	92
5.2	Molding of Two-stages Motion Perception System	93

5.2.1	First-stage: Local Motion Energy Computation	93
5.2.2	Second-stage: Global Motion Integration and Segregation	96
5.2.3	Training Strategy	99
5.3	Analyses	100
5.3.1	In silico Neurophysiological Study	100
5.3.2	Psychophysical Analysis	104
5.4	Discussion on Unsupervised Learning	112
5.5	Methods and Implementation details	113
5.5.1	Dataset	113
5.5.2	Training	115
5.5.3	In Silico Neurophysiological Test	115
5.6	Discussion and Conclusion	118
6	Human-aligned Visual Motion Processing Model: Second-order Motion	121
6.1	Introduction	122
6.2	Results	125
6.2.1	Two-stages Processing Model	126
6.2.2	Materials Properties & Second-order Motion Perception	127
6.3	Discussion	130
6.4	Methods	137
6.4.1	Model Structure	137
6.4.2	Training Strategy	137
6.4.3	Datasets Generation	138
6.4.4	Expeirmental Details	139
7	Conclusion	143
7.1	Overview	143
7.2	Limitations and Future studies	145
7.2.1	Limits of Current Unsupervised Learning	146
7.2.2	Unclear Neural Implementation	148
7.2.3	Blacks Box Problem Still Exists	149
7.2.4	Vision Functionality is Interconnected	150
	Bibliography	151
	List of Publications	167

List of Figures

1.1	Examples of Exner’s neural pathway assumption and Reichardt detector	3
1.2	Illustration of optical flow in psychology and computer vision	5
2.1	Three descriptive methods of motion	11
2.2	Receptive field of simple cells in V1.	14
2.3	Description of motion and the ideal motion detector in Fourier space.	17
2.4	Motion energy model	18
2.5	Illustration of first and second order motion (Drift-balanced motion) in spatiotem- poral and Fourier space.	20
2.6	Aperture problem.	22
2.7	Motion integration using grating and paid	23
2.8	Demonstration of common optical flow estimation datasets	25
2.9	Demonstration of the encoder-decoder architecture of FlowNet	27
2.10	The architecture of the spatial pyramid network	28
2.11	Pyramid network for optical flow estimation	29
2.12	Demonstration of GMflow	30
3.1	Graph summary of temporal dynamic scene augmentation.	38
3.2	Demonstration of spatiotemporal adaptive fusion	39
3.3	Inference architecture of the spatial-temporal dual recurrent block	42
3.4	Occlusion-aware temporal smoothness regularization	44
3.5	Mixed supervision strategy with dynamic occlusion	47
3.6	Qualitative demonstration of temporal dynamic variations	49
3.7	Complete training pipeline based on self-supervised learning	51
3.8	Visualization of results on Sintel and KITTI datasets	55
3.9	Network EPE performance with increasing numbers of inference frames	58
3.10	Correspondence entropy distribution	61
3.11	Global correspondence matching block (GCM)	62
3.12	Whole inference pipeline of the spatiotemporal adaptive fusion network	65
3.13	Simulation of dynamic occlusion and with the pseudo label	68
3.14	Quantitative demonstration on DAVIS dataset	72

4.1	Responses of several computer vision models	80
4.2	Responses of RAFT trained on different datasets	81
4.3	Responses on classical motion stimuli & illusion	83
4.4	Barber pole illusion	84
4.5	Evaluation of motion integration using plaid stimuli	85
4.6	Test on global motion integration	87
4.7	Motion integration with recurrent process	88
5.1	Molding of motion perception system, Stage I	94
5.2	Molding of motion perception system, Stage II (A)	98
5.3	Molding of motion perception system, Stage II (B)	99
5.4	Properties of trained complex cells in stage I	101
5.5	In silico neurophysiological test: directional tuning	102
5.6	In silico neurophysiological test: speed tuning	103
5.7	Missing fundamental illusion	104
5.8	Psychophysical stimulus tests motion integration.	106
5.9	Mario reverse-phi illusion	107
5.10	Barber pole illusion	107
5.11	Performance on complex natural scenes	108
5.12	Demonstration on Sintel Scenes	110
5.13	Qualitative comparison between our model and other related models	111
5.14	Qualitative result of unsupervised learning methods	112
6.1	The proposal of dual channel structure in the previous study	123
6.2	The optical turbulence on the surface with non-diffuse materials	124
6.3	Two-Stages motion perception system prototype	126
6.4	Material-controlled natural motion dataset	128
6.5	Second-order benchmark demonstration	129
6.6	Human response on second-order motion	130
6.7	Interplay between material properties and second-order motion perception	131
6.8	Demonstration of the interplay between material properties and second-order motion perception	132
6.9	The roles between first-order and higher-order motion channels	134
6.10	Second-order motion in natural environment	135
6.11	The pipeline of the human psychophysical experiment.	140

List of Tables

3.1	Comparison on the Sintel Benchmark	53
3.2	Comparison on KITTI Benchmark	54
3.3	Main ablation study on different combinations of the proposed components. . . .	57
3.4	Ablation study on model architectures	59
3.5	Ablation study using different combinations of the three training enhancers. . . .	59
3.6	Ablation study comparing dynamic scene variation to static data augmentation.	60
3.7	Main ablation study on different combinations of the proposed components . . .	71
3.8	Ablation study on LMC metrics	73
3.9	Ablation study of LMC augmentation.	73
3.10	Comparison with previous methods on the Sintel Benchmark	74
3.11	Comparison on KITTI 2012 Benchmark	75
3.12	Comparison with prototypes in supervised learning	76
4.1	Summary of different motion estimation models	79
5.1	Model v.s. Human v.s. Ground truth	110

Glossary

AI Artificial Intelligence. 91

BSDF Bidirectional scattering distribution function. 124

CNNs Convolutional neural networks. 27

DNNs Deep neural networks. 6

GRU Gated recurrent unit. 42

IoC Intersection over union. 22

LCD Liquid crystal display. 139

LGN Lateral geniculate nucleus. 13

LMC Low matching confidence. 37

MFFC Motion from Fourier component. 3

MST Medial superior temporal area. 24

RF Receptive field. 4

SoTA State-of-the-art. 7

V1 Striate visual cortex. 3, 13

VA Vector Average. 22

Chapter 1

Introduction

Vision is essential for humans and other animals to survive in this world. Throughout our long history, humans have evolved a sophisticated visual system capable of handling various tasks with remarkable stability and generalization across different visual environments. Creating machines that perceive the world as humans do poses a significant research challenge with profound implications for cognitive science and engineering. In cognitive science, the aim is to develop image-computable models of human perception and cognitive systems to understand better how the brain functions internally. In engineering, machines equipped with human-like capabilities could enhance the reliability and stability of various human-centered technologies.

Among the various functions of vision, visual motion estimation is crucial for humans to perceive and interact with the dynamic world. In the human visual system, this function generally involves inferring the speed and direction of elements within a scene, utilizing inputs from visual, vestibular, and proprioceptive sources. Although this process might seem straightforward to most observers, it represents a complex computational challenge and is difficult to explain regarding neural processing. In computer vision, visual motion estimation is quantified as optical flow (2D) or scene flow (3D) within the image coordinates between two frames. Optical flow is typically described by a 2D vector (u, v) that illustrates the pixel correspondence or shifts between two consecutive frames in a video, reflecting the movement of 3D objects in world coordinates projected onto 2D image coordinates. Scene flow, on the other hand, extends this concept to three dimensions by incorporating depth information, resulting in a 3D vector field (u, v, w) that captures the full motion of points in 3D space. The applications of this function are extensive in the industry, including object tracking, predicting future states in autonomous driving [112], environmental perception for intelligent robots [54], and video interpolation/ super-resolution [72] in image and video editing. As a result, visual motion estimation has garnered significant attention across multiple research fields, including optical flow/ scene flow estimation in image processing and computer vision, as well as studies of visual motion perception in psychology and neuroscience.

1.1 Object of This Study

This study explores visual motion perception across these research domains, focusing on the computational modeling of human-aligned visual motion processing systems. Although different fields may have their own methodologies and sub-goals, the long-term objective is to establish a general computational model that mimics human perception. This is not only crucial for a deeper understanding of the mechanisms behind human visual motion processing systems but also offers significant benefits for downstream applications in society.

The goal of developing a human-like machine vision model goes beyond merely achieving technical accuracy in tasks like optical flow estimation. While current machine vision systems surpass human vision in terms of precision and speed under controlled conditions, there are compelling reasons to mimic human vision:

- **Scientific Understanding:** From a scientific perspective, we seek to comprehend the workings and internal mechanisms of the human visual system. Using modeling to explain and validate these mechanisms is a common approach in vision science, making it meaningful on a scientific level. By building a human-aligned model, we will show that we gain insights similar to those derived from understanding the functionality and mechanisms of first-order and second-order motion perception in human vision systems.
- **Innovation Inspired by Human Vision:** The human motion perception system can inspire applications, such as its unique dual-channel mechanism for processing first-order and second-order motion. We will show these human insights potentially offer resilience against chaotic, high-optical-noise environments, possibly leading to a more stable and reliable visual motion estimation model. Current studies in the optic flow domain demonstrate that existing models can be easily disrupted by certain adversarial attacks, highlighting the need for models inspired by the robustness of human vision.
- **Human-Machine Interaction:** As technology becomes more integrated into everyday life, enhancing the interaction between humans and machines is increasingly crucial. Developing machine vision systems that process and respond to visual information in a human-like manner can make these interactions more intuitive and natural. This is especially important in areas such as assistive technology, where machines must adapt to human behaviors and preferences.

With the ultimate goal of modeling a human-aligned model in mind, this Chapter will first provide a general introduction and conceptual overview of the modeling motion processing systems in the fields of vision science and computer vision.

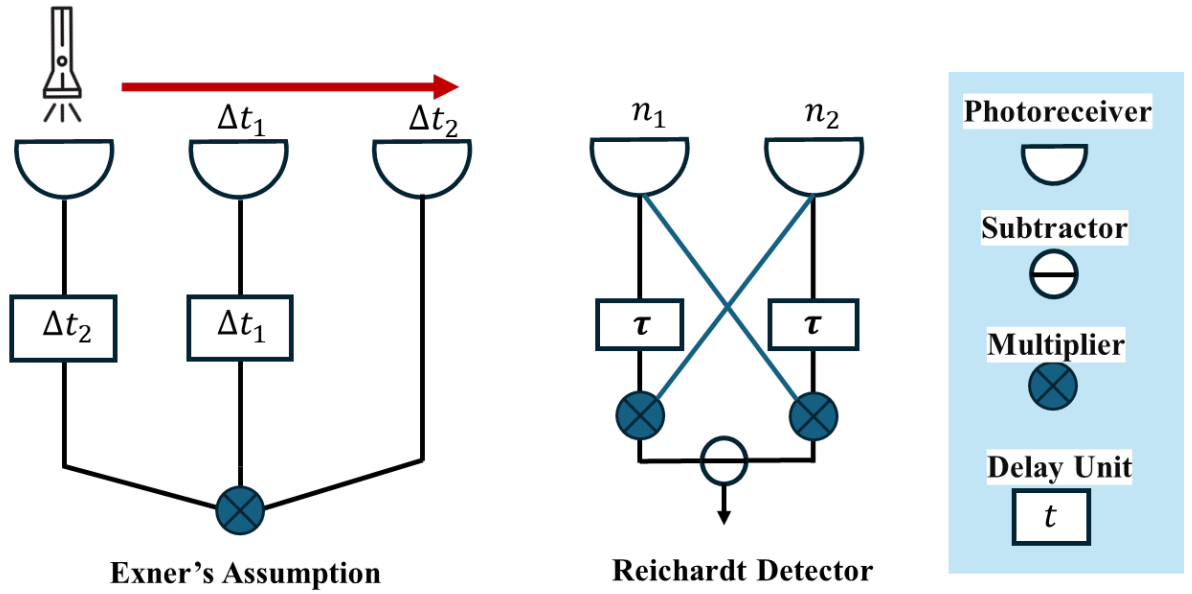


Figure 1.1: Examples of Exner's neural pathway assumption (Left [41]) and Reichardt detector (Right [57]).

1.2 Modeling Motion Perception in Vision Science

Research into motion perception began with Exner's [41] discovery of "apparent motion," where a light flashed in one position and then in another nearby position, creating the illusion of motion without actual movement. This phenomenon led to the identification of neural pathways sensitive to motion direction. Hassenstein and Reichardt [57] later developed a computational model, known as the Reichardt-Hassenstein detector, which used photoreceptors to detect motion through delayed inhibition between two photoreceptors over space and time, illustrated in Figure 1.1.

The Reichardt detector laid the groundwork for understanding biological motion perception. Unlike insect compound eyes, primate visual systems have cells in the striate visual cortex (V1) that detect orientation and spatial frequency of luminance, known as simple cells. These cells linearly sum responses across their receptive fields to generate Gabor-like fields suitable for detecting luminance changes, leading to the development of various motion perception models. These models, including the Reichardt model and the motion energy model [2], derive motion information from luminance changes over time and are known as first-order motion models or motion from Fourier components (MFFC) principles.

Higher-order Motion Perception

First-order motion models cannot explain all motion phenomena. In 1970, Henning, Hertz, and Broadbent [59] first introduced second-order stimuli based on contrast modulation, which

remain invisible to first-order detectors. Later, the second-order motion was first demonstrated by Derrington and colleagues [8, 36] and further explored by various studies like Sperling [25, 157], Zanker [197], Nishida [120, 124, 125], Mather [101, 103] and others. These stimuli involve attributes like spatiotemporal contrast and spatial density rather than conventional motion defined by a luminance change, which contains no energy in the Fourier plane describing the direction of perceived motion, has variously been called “non-Fourier motion,” “second-order motion,” “higher-order motion,” and sometimes “feature motion.” For simplicity, we refer to it as second-order motion in this dissertation.

It is suggested that second-order motion is processed by different visual mechanisms than first-order motion, with research indicating separate encoding systems. Studies support this distinction, including psychophysical investigations [120, 124], brain injury cases, and neurophysiological experiments. For example, Mather et al. [101–103] found different processing mechanisms for first- and second-order motions, while Baker’s electrophysiological studies showed distinct neural responses to these motions [9, 10].

Current models involve early-stage filtering and nonlinear operations to extract the feature related to second-order motion [154] like Smith [154], and Sperling [96] introduced a model that added an additional feature extraction process to the general first-order and second-order channels. Baker’s model employs a filter-rectify-filter approach to process second-order stimuli [199].

Motion Integration

The previous discussion on local motion detection highlights its fundamental role in the early stages of visual motion processing. However, accurately representing the motion of objects in the visual field involves a more complex problem: the integration of information from elementary motion detectors.

Due to the limited receptive field (RF), neurons in V1 only detect motion perpendicular to an object’s edge, known as the aperture problem, which requires the integration of motion vectors from multiple contours. Studies on area MT neurons show they integrate multiple motion directions, responding to preferred pattern directions even without individual components moving that way. Although several computational models of area MT have been proposed [152, 184], a more comprehensive computational architecture that replicates the motion integration abilities of MT neurons and human perception, especially the human perception in complex natural scenarios, remains elusive [129].

1.3 Motion Estimation in Computer Vision

In computer vision, motion estimation is generally represented by optical flow (2D) and scene flow (3D).

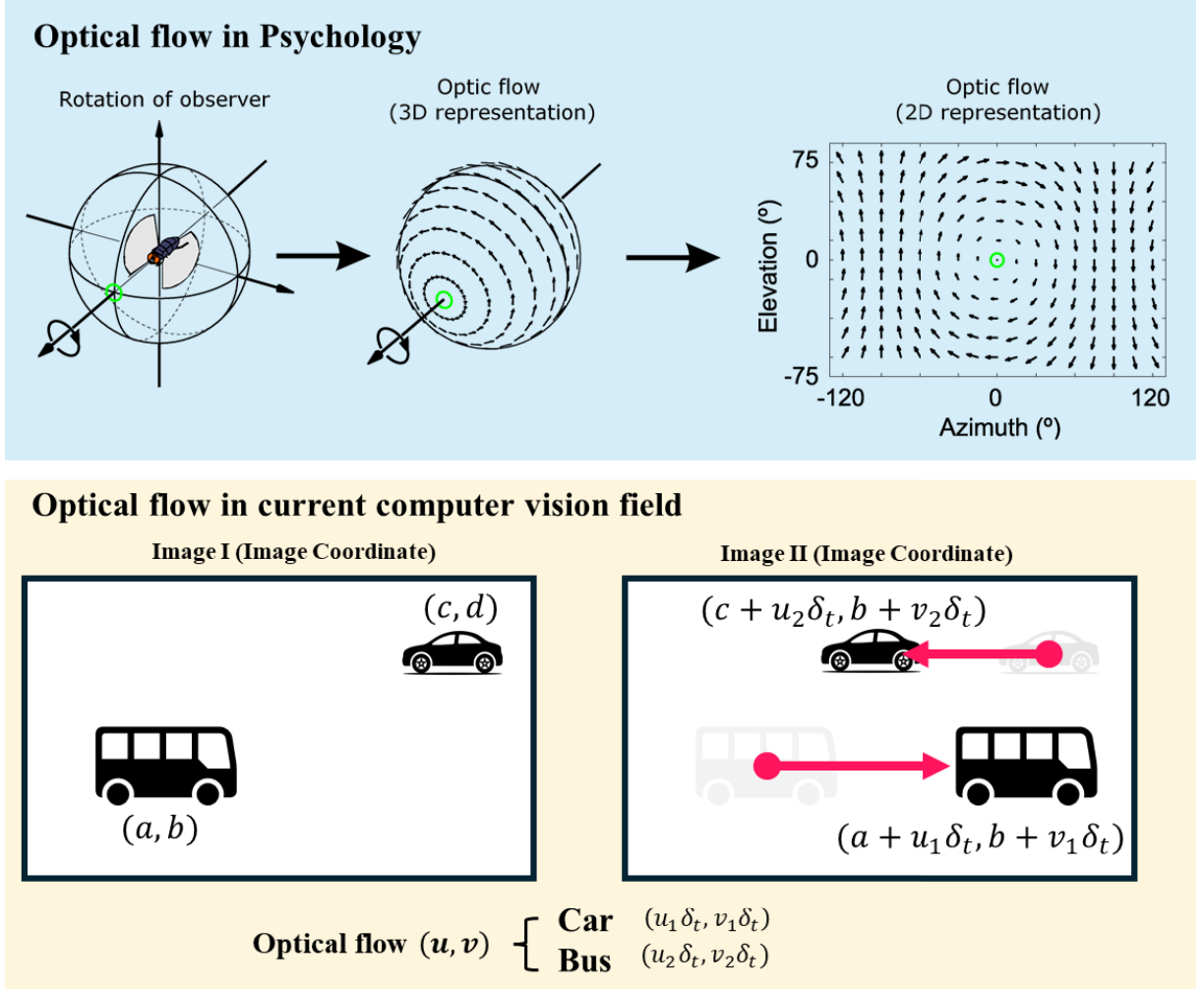


Figure 1.2: Illustration of optical flow in psychology and computer vision. The top figure is adapted from PloS Biology (CC licensed) [49].

The concept of optical flow was first introduced by Gibson [49] to describe the visual stimuli experienced by animals as they move through their environment. This concept has since been adopted across various disciplines, and in the current field of computer vision, it refers to the displacement of objects in three-dimensional space projected onto two-dimensional pixel coordinates in a sequence of images, representing the objects' instantaneous motion in terms of speed and direction. Quantitatively, it is represented as a two-dimensional vector field (u, v) , known as the optical flow field, as illustrated in Figure 1.2.

Scene flow, on the other hand, represents the 3D motion of points in a scene, extending the concept of optical flow to three dimensions by capturing the motion in 3D space. It is depicted as a 3D vector field (u, v, w) , where u and v are the components of motion in the image plane, and w represents the motion in the depth direction. Although scene flow provides a more comprehensive representation of motion by including depth information, studies in human vision science primarily focus on 2D perception compared to 3D [121, 123]. The primary objective of

this thesis is to develop a human-aligned motion perception model, and our focus is on the perception of 2D motion because humans can perceive motion without directly considering depth cues, and 2D motion perception remains a critical area for understanding human motion perception. Therefore, this thesis begins with an investigation of 2D motion and primarily discusses the issues of 2D optical flow.

Historically, optical flow estimation methods focused on minimizing complex energy equations [60, 198]. However, these methods were hampered by slow inference speeds, an inability to produce dense optical flows, and limitations in handling complex scenarios. The advent of deep learning has significantly advanced optical flow techniques. FlowNet [38] pioneered the use of CNNs for dense optical flow estimation, leading to the development of various approaches based on deep neural networks (DNNs). Current dense optical flow estimation methods are primarily divided into supervised and unsupervised learning approaches.

Unlike other vision tasks, where ground-truth labels for dense optical flow are rarely available for real image pairs, supervised optical flow techniques are typically trained using synthetic data. This results in an inherent mismatch between real and synthetic data, leading to a disparity in data distribution that needs to be addressed to improve performance [104]. The limitations of synthetic data enhance the viability of unsupervised optical flow estimation.

Unsupervised learning frameworks use unlabeled videos [193], deriving a photometric loss that measures the difference between the target image and the (inversely) warped source image based on the dense flow field predicted by the network. However, image-warping-based unsupervised learning faces multiple challenges, including brightness and color variations, motion blur in dynamic environments, and spatial occlusion caused by moving objects, which leads to pixel correspondence loss over frames. These factors mislead the networks and degrade performance. Although several studies have attempted to solve these problems by designing more robust loss functions [69, 110], using masks to exclude occluded regions [182], and tracking occluded pixels in adjacent frames [182], these challenges remain significant for unsupervised learning methods.

1.4 Current Problems and the Organization

Based on the current situations in visual motion estimation and perception, this thesis indicates several existing gaps and problems between these two areas.

In vision science, many studies try to understand and explain the mechanism behind the visual motion perception system, and several computational models are proposed to mimic human perception and the underlying neuron activation patterns. Nonetheless:

- Current studies already establish the well computational structures to illustrate the simple neurons in the V1 area and some basic functions such as the direction tuning and speed tuning ability, and by using the drifting grating or plaid, these kinds of models are able to explain the perception in experimental environments. However, an integrated model that

is capable of dealing with more complex scenarios beyond the simple environment setting, like natural scenarios, still lacks investigation.

- Especially, the function of MT region, which is important to link the local motion energy from simple cells and get the flexible interpretation on complex environments; several computational models of area MT have been proposed, but mechanisms that can fully encapsulate a variety of motion integration abilities of MT neurons and/or human perception have yet to be developed.
- Beyond the fundamental topic of first-order motion, motion perception involves more high-level processing, such as second-order motion. In the past, second-order motion perception has been extensively studied [100] [10] using carefully designed artificial stimuli (e.g., drift-balanced motion [25]) to control first-order motion components, but why and how the visual system has acquired this perceptual ability in natural environments remains poorly understood.

In computer vision, recent advances in deep learning have propelled machine vision capabilities to a level comparable to or surpassing human performance in specific vision tasks [35, 93]. Visual motion estimation [38] is not an exception, where state-of-the-art (SoTA) CV models based on deep learning have demonstrated superior accuracy in optical flow estimation compared to human estimation [189]. However:

- Animals use an unsupervised-like learning method to perceive motion in unlabeled real scenes. However, current unsupervised learning methods in computer vision still face challenges when pixel correspondence is distorted due to occlusion, motion blur, brightness variation, etc. Additionally, these methods primarily decompose the motion into a discrete case that relies on two-frame training and inference, whereas animals learn and perceive motion in a continuous world.
- The current CV models fail to replicate the full spectrum of human motion perception. For example, CV models are often unstable under various experimental conditions [141, 167], fail to reproduce human visual illusions, and do not fully capture the biases inherent in human perception [189].
- In structural design, the computer vision model didn't explicitly align the neurobiological plausibility and human mechanism used in the calculation. Most of their tasks focus on correspondence searching between pairs of frames to deduct the optical flow. They are trying to find a solution to maximize object correspondence, which is not explicitly carried out by humans as motion is a sensation, not correspondence matching.
- Studies have demonstrated that various species, such as macaques [128], flies [172], and humans [10], can perceive motion through high-level image features like contrast and

temporal modulations, such motion known as second-order motion (non-Fourier motion) is a sub-type of higher-order motion. Present-day machine vision models, particularly those focusing on optical flow estimation [38, 62, 171], overlook this aspect.

To address the problem listed above, this thesis is organized as follows:

- Chapter 2 provides preliminaries associated with visual motion perception and optical flow estimation tasks. Specifically, it describes the current methodology of supervised and unsupervised learning for optical flow and identifies potential problems. Concurrently, it introduces basic computational methods for motion perception in both CV fields—such as the Lucas-Kanade method, DNNs-based methods, and in VS fields, which include models based on Motion From Fourier Component (MFFC) principles [25] and higher-order motion perception such as second-order motion perception.
- Chapter 3 will explore solutions for the unsupervised learning of motion perception. We will discuss the challenges faced by current unsupervised learning methods, such as occlusion, which distorts the correspondence between frames. Considering that motion is a smooth dynamic process, and that humans and animals learn to perceive motion across continuous frames, we propose a solution based on modeling unsupervised optical flow from multi-frame sequences of dynamic scenes. We introduce a novel temporal recurrent model that accumulates motion history and leverages temporal continuity across frames to address content variation and occluded regions.
- Chapter 4 will show the difference between CV models and human perception in motion perception tasks, from their inner mechanism to the perceptual response. We incorporated several representative CV models, including supervised and unsupervised methods, and compared them with a human perceptual response based on the in-silico psychophysical method. We will show that most of the DNN-based models did not generalize to simple experimental stimuli, and DNN-based models cannot account for perceptual motion reversals perceived by humans with contrast-reversed patterns and a missing fundamental grating. In addition, none of the tested models can explain occlusion-aware motion integration and second-order motion.
- Chapter 5 proposes an image-computable model of human motion perception by bridging the gap between biological and CV models. Specifically, it introduces a novel two-stage approach that combines trainable motion energy sensing with a recurrent self-attention network for adaptive motion integration and segregation. This model architecture aims to capture the computations in V1-MT, the core structure for motion perception in the biological visual system, while providing the ability to derive informative motion flow for a wide range of stimuli, including complex natural scenes.

- In Chapter 6, the issue of second-order motion and its meaning in natural environments are discussed. It proposes a DNN-based model to process both first- and second-order motions in natural scenes. By creating initial second-order motion benchmarks to quantify the ability of second-order motion and training the model using specific optical flow datasets with varying material properties, we observed that second-order motion perception is significantly related to the material properties of training environments. This suggests that the second-order motion system might have evolved, at least partially, to help in the robust estimation of object motion while countering optical turbulence under natural environments.
- In the final Chapter 7, we will conclude the contribution and findings of this dissertation, joint with and discuss the limitations and the future extension in this field.

Chapter 2

Preliminary Background

This chapter provides preliminaries associated with visual motion perception and optical flow estimation tasks, which serve as the solid basis for studies in the following chapters.

Firstly, we will introduce the background of motion perception in the human visual field, covering early motion processing mechanisms from neurophysiological studies and related computational models such as the Reichardt detector, motion energy models, motion from Fourier component (MFFC) principles, and higher-order motion processing. These concepts establish a solid foundation for developing a human-aligned model in Chapter 5 and Chapter 6.

Secondly, we will examine current methodologies for motion estimation, with a particular focus on optical flow estimation in the computer vision field. We will discuss several representative model structures like FlowNet, PWCNet, and GMFlow, along with general unsupervised learning methods based on image-warping and reconstruction technologies. This background will be integrated into Chapter 3 to develop human-motivated unsupervised learning methods. Additionally, the CV models introduced in this chapter will be revisited in Chapter 4, Chapter 5, and Chapter 6 for comparison with human perception and our proposed methods.

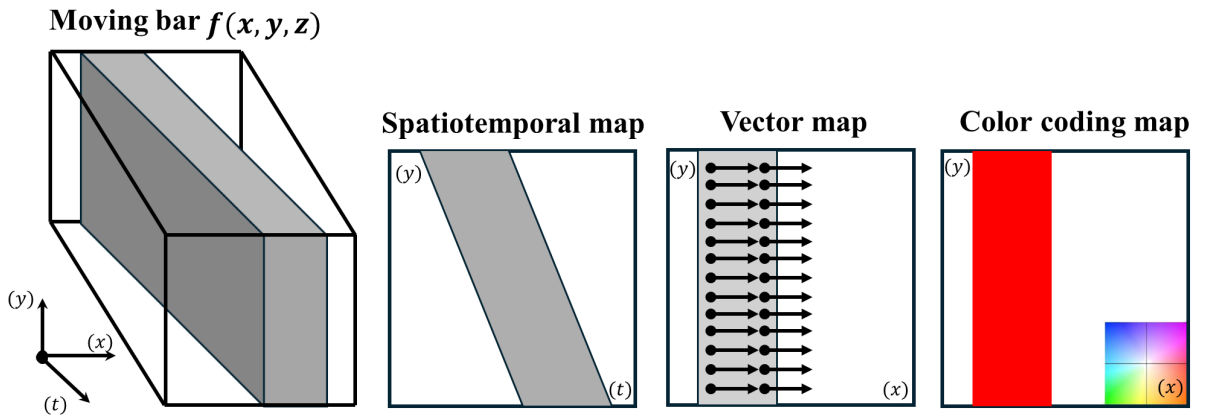


Figure 2.1: Three descriptive methods of motion. The left figure is redrawn from the [2].

2.1 Description of Motion

In a natural environment, motion is a process defined simultaneously by time and space, so some conventions are needed to represent it in a static image. Let's explore this concept with a simple example depicted in the diagram: a vertically oriented bar moving to the right, as shown in Figure 2.1. We can perceive the motion of visual images as events correlated through time and space, which can be mathematically represented by a function $I(x, y, t)$. Here, x and y denote the horizontal and vertical spatial dimensions, respectively, while t represents the temporal dimension. Consequently, the bar in the illustration occupies a three-dimensional spatiotemporal space, where three characters of motion can be defined:

1. **Offset** is defined as the change in location of this bar over a given period of time. It can be broken down into spatial and temporal components.
2. **Spatial Component of the Offset**: the displacement vector between two positions in two-dimensional space (x_1, y_1) and (x_2, y_2) . The vector indicates the direction of the motion.
3. **Temporal Component of the Offset**: the time (t) required to traverse the distance indicated by the displacement vector. The speed of the motion is determined by dividing the length of the displacement vector by the time taken to travel that distance.

To represent these characters of motion using static figures, we adopt three methods to represent motion. Firstly, we can add vector information to a specific point to indicate the direction and magnitude of its movement. Secondly, a common method is to use color coding of the motion vector, thus mapping the vector into continuous HSV color space; this method is widely adopted in many computer vision approaches. Lastly, using a spatiotemporal coordinate system can reflect motion information along a particular dimension. For instance, in the space-time plot shown, the boundaries of a translating bar accumulate into a slanted trajectory over time. A slant toward the positive x-axis indicates rightward movement, while a slant toward the negative x-axis indicates leftward movement. The degree of the slant corresponds to the speed of the motion. Similar descriptions can also be applied to vertical movements. Space-time plots objectively reflect the information of motion, facilitating theoretical analysis. This thesis will alternately use these three methods to describe motion.

2.2 Early Findings in Motion Perception

Research into motion perception dates back to important discoveries by Exner [41], who devised a stimulus where a light flashed in one position, then briefly afterward in a neighboring position. Although there was no actual movement of the target, observers still perceived motion, a phenomenon known as "apparent motion." From in-depth studies of this phenomenon, the neural

pathways sensitive to motion direction were first described. Figure 1.1 (left) illustrates a neural pathway sensitive to rightward translational motion. In the diagram, when the light moves to the right, the neuron that is activated first has a longer synaptic delay due to its longer synaptic connection. The neighboring neuron has a shorter delay, allowing all responses to almost simultaneously reach the output neuron. This synchronous activation ensures the excitation of the output neuron. Conversely, when the light moves from right to left, responses arrive asynchronously at different times at the output neuron, preventing it from reaching sufficient excitation.

In subsequent research, Hassenstein and Reichardt [57] discovered similar behavioral response characteristics in the visual systems of certain insects, leading to the development of the first computational model of motion perception, known as the model of Reichardt-Hassenstein detector. This initial model consists of two symmetrical, mirrored branches tuned to opposite directions of motion, as illustrated in Figure 1.1. In the model, photoreceptors $\{n_1, n_2\}$ directly interact with spatial luminance sampling data, performing multiplication operations to correlate inputs from different moments. Specifically, the left branch multiplies the signal at one spatial position with the signal at an adjacent position after a delay. Conversely, the right branch multiplies the signal at another spatial position with the delayed signal from a further position. The difference in outputs between the two branches indicates the direction of motion. The time delay and the comparative multiplication operation between the branches enable the model to reflect the degree of match between input signals. The Reichardt model can accurately predict various types of motion stimuli, including complex waveforms and some counterintuitive phenomena. According to this model, motion is simply perceived through delayed inhibition between two photoreceptors over space and time. Evidence for a similar mechanism has also been confirmed in rabbits and other mammals [11, 12].

The Reichardt detector provides a fundamental computational model for biological visual motion processing systems. However, unlike the compound eyes of insects, the receptive fields of cells in the primate visual system are considered to have specific characteristics such as frequency and orientation. These cells are located in the striate visual cortex (V1), specifically referred to as simple cells in V1, which typically have small, elongated receptive fields and are thought to perform narrow-band selection for orientation and spatial frequency of luminance information, as illustrated in Figure 2.2.

In a simplistic model, the simple cell linearly sums responses across their receptive field areas, generating the Gabor-like receptive fields, which are suitable for detecting boundaries characterized by luminance changes. Maximum response output occurs only when the pattern of light and dark areas matches the excitatory and inhibitory spatial structures of the receptive fields. As illustrated in Figure 2.2 (Left), neurons in the lateral geniculate nucleus (LGN) are arranged in a certain pattern.

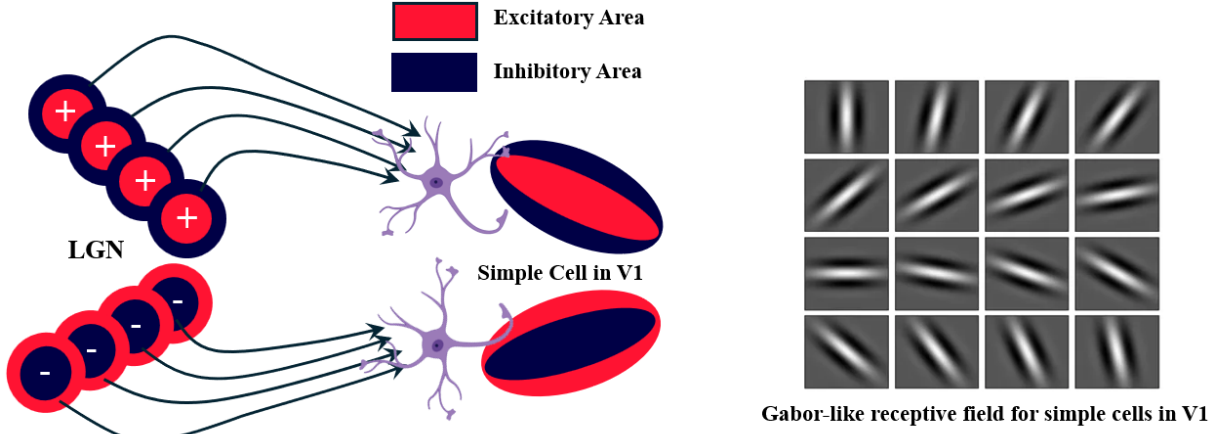


Figure 2.2: Receptive field of simple cells in V1.

2.3 Modelling Visual Motion Processing

2.3.1 Local Motion Detector

Spatiotemporal Gradients

For motion, an intuitive approach is to define motion as the change in the position of an object in space over time. This understanding posits that observers perceive motion when the brightness of an object varies simultaneously in both time and space. To detect local motion, changes in both time and space must be considered together.

In 1981, Horn and Schunck introduced a pioneering method for estimating optical flow based on a variational framework [60]. They constructed an energy function incorporating prior knowledge constraints, notably the global brightness constancy assumption and the spatial smoothness assumption. By iteratively optimizing this energy function, a dense optical flow field could be estimated. This approach of using variational methods for optical flow estimation gained widespread popularity. Shortly thereafter, Lucas and Kanade [97] proposed a method that incorporated local neighborhood constraints. This method, focusing on localized regions, enabled the generation of sparse optical flow fields. Such classical methods typically rely on the brightness constancy constraint [60] (i.e., $I(x, y, t) = I(x + dx, y + dy, t + dt)$), associating spatiotemporal information from different spatial locations at different times, where a voxel at location (x, y, t) with intensity $I(x, y, t)$ have moved by dx, dy, dt between two frames.

Assuming the sampling rate is sufficient, the image constraint at $I(x, y, t)$ can be expressed using a Taylor series expansion:

$$I(x + \Delta x, y + \Delta y, t + \Delta t) = I(x, y, t) + \frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t + \dots$$

By truncating the higher-order terms (linearization), we obtain:

$$\frac{\partial I}{\partial x} \Delta x + \frac{\partial I}{\partial y} \Delta y + \frac{\partial I}{\partial t} \Delta t = 0$$

Dividing by Δt gives:

$$\frac{\partial I}{\partial x} \frac{\Delta x}{\Delta t} + \frac{\partial I}{\partial y} \frac{\Delta y}{\Delta t} + \frac{\partial I}{\partial t} \frac{\Delta t}{\Delta t} = 0$$

This leads to:

$$\frac{\partial I}{\partial x} V_x + \frac{\partial I}{\partial y} V_y + \frac{\partial I}{\partial t} = 0 \quad (2.1)$$

where (V_x, V_y) are the x and y components of the velocity or local optical flow of $I(x, y, t)$, and $\frac{\partial I}{\partial x}$, $\frac{\partial I}{\partial y}$, and $\frac{\partial I}{\partial t}$ are the derivatives of the image in the respective directions at (x, y, t) [97].

This is one equation with two unknowns (V_x, V_y) , which cannot be directly solved. This is known as the **Aperture Problem** in optical flow algorithms. The supplementary conditions or constraints are needed to estimate the actual flow.

Motion Analysis in Fourier domain

The aforementioned algorithm provides a fundamental concept for motion computation, which treats motion as a spatiotemporal event described by the brightness function of spatial position and time. On the other hand, the Fourier analysis of the motion pattern, i.e., the frequency domain, also contains useful motion information. Consider a rigid movement at a constant velocity. Its brightness function can be represented as:

$$I(x, y, t) = I(x - v_x t, y - v_y t)$$

where v_x and v_y are the horizontal and vertical velocity components, respectively. The Fourier transform of this function is given by:

$$\begin{aligned} I(u, v, w) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(x, y, t) e^{-i2\pi(xu + yv + wt)} dx dy dt \\ &= \iiint I(x - v_x t, y - v_y t) e^{-i2\pi(xu + yv + wt)} dx dy dt \end{aligned}$$

Let $x_1 = x - v_x t$ and $y_1 = y - v_y t$, then:

$$dx_1 = dx, \quad dy_1 = dy, \quad dt = dt$$

The integral becomes:

$$\begin{aligned} I(u, v, w) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} I(x_1, y_1) e^{-i2\pi((x_1 + v_x t)u + (y_1 + v_y t)v + wt)} dx_1 dy_1 dt \\ &= \iiint I(x_1, y_1) e^{-i2\pi(x_1 u + y_1 v + wt + v_x t u + v_y t v)} dx_1 dy_1 dt \\ &= \iint I(x_1, y_1) e^{-i2\pi(x_1 u + y_1 v)} dx_1 dy_1 \int e^{-i2\pi(w + v_x u + v_y v)t} dt \end{aligned}$$

The integral with respect to t is a Dirac delta function:

$$\int e^{-i2\pi(w + v_x u + v_y v)t} dt = \delta(w + v_x u + v_y v)$$

Thus, the Fourier transform is:

$$I(u, v, w) = I(u, v) \delta(w + v_x u + v_y v)$$

Combining the properties of the Dirac delta function, the energy of the motion will only concentrate when $w + v_x u + v_y v = 0$. This means that the spectrum of the translated image is a plane passing through the origin, with its direction corresponding to the object's velocity.

More specifically, let's consider a 45-degree inclined cosine wave moving uniformly with velocity c . Its brightness function can be represented as:

$$I(x, y, t) = \cos\left(\frac{k}{\sqrt{2}}(x + y - 2ct)\right)$$

Where k is the spatial frequency and the wave is moving at a constant velocity c such that $v_x = v_y = c$. Next, we compute the Fourier transform of $I(x, y, t)$:

$$\begin{aligned} I(u, v, w) &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \cos\left(\frac{k}{\sqrt{2}}(x + y - 2ct)\right) e^{-i2\pi(xu + yv + wt)} dx dy dt \\ &= \frac{1}{2} \iiint \left[e^{i\left(\frac{k}{\sqrt{2}}(x + y - 2ct)\right)} + e^{-i\left(\frac{k}{\sqrt{2}}(x + y - 2ct)\right)} \right] e^{-i2\pi(xu + yv + wt)} dx dy dt \end{aligned}$$

We handle each exponential term separately. For the first term:

$$\begin{aligned} &\iiint e^{i\left(\frac{k}{\sqrt{2}}(x + y - 2ct) - 2\pi(xu + yv + wt)\right)} dx dy dt \\ &= \iiint e^{i\left[\left(\frac{k}{\sqrt{2}} - 2\pi u\right)x + \left(\frac{k}{\sqrt{2}} - 2\pi v\right)y - \left(2\pi w + 2\frac{kc}{\sqrt{2}}\right)t\right]} dx dy dt \end{aligned}$$

The Fourier transform gives us Dirac delta functions:

$$= \delta\left(u - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(v - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(w + \frac{kc}{\pi\sqrt{2}}\right)$$

Similarly, for the second exponential term $e^{-i\left(\frac{k}{\sqrt{2}}(x + y - 2ct)\right)}$:

$$\begin{aligned} &\iiint e^{-i\left(\frac{k}{\sqrt{2}}(x + y - 2ct) + 2\pi(xu + yv + wt)\right)} dx dy dt \\ &= \delta\left(-u - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(-v - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(-w - \frac{kc}{\pi\sqrt{2}}\right) \end{aligned}$$

Combining these results, the final Fourier transform of the uniformly moving sinusoidal wave is:

$$\begin{aligned} I(u, v, w) &= \frac{1}{2} \left[\delta\left(u - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(v - \frac{k}{2\pi\sqrt{2}}\right) \delta\left(w + \frac{kc}{\pi\sqrt{2}}\right) \right. \\ &\quad \left. - \delta\left(u + \frac{k}{2\pi\sqrt{2}}\right) \delta\left(v + \frac{k}{2\pi\sqrt{2}}\right) \delta\left(w - \frac{kc}{\pi\sqrt{2}}\right) \right] \end{aligned} \tag{2.2}$$

The equation 2.2 indicates the moving sinusoidal grating has a pair of energy concentration points asymmetrically located in the 3-D Fourier space, the location of which clearly indicates the moving speed and direction. The computational model analyzes such Fourier information to drive the motion estimation and the spatiotemporal Gabor filter is one of the representative local motion detectors in the frequency domain. Also, in neuropsychology, the responses of simple cells turned out to be captured by Gabor filters. That simple cells act like Gabor filters makes sense since the ideal Gabor filters are known to be optimal in achieving maximal resolution

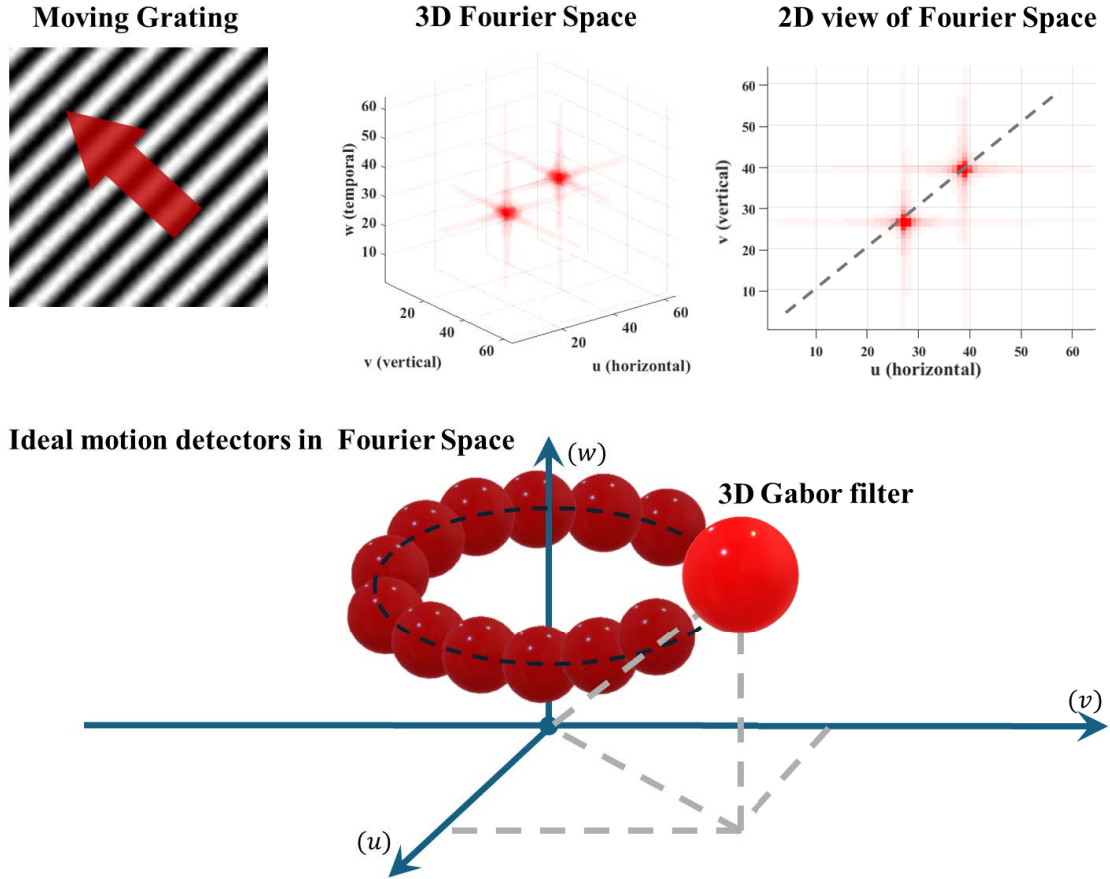


Figure 2.3: Description of motion and the ideal motion detector in Fourier space.

in both the spatiotemporal and the associated frequency. Consequently, they require minimal filters to represent spatiotemporal information, as demonstrated in Figure 2.3.

Motion Energy Model

Based on the MFFC principles, Adelson and Bergen [2] designed a motion energy computation model, which became the standard reference model for low-level, Fourier-based motion sensing in the human visual system.

The motion energy model equates motion to spatiotemporal orientation; in a spatiotemporal plot, a moving edge has an inclined trajectory and can, therefore, be detected by sensors with spatiotemporal directionality. As shown in Figure 2.4 (A), stimuli are represented in x - t space. The image of a moving bar can be plotted in a three-dimensional volume that has two spatial dimensions (x and y) and a temporal dimension (t). A two-dimensional cross-section through this volume along a fixed y -position produces an x - t plot; x -position as a function of time. Movement creates an oriented structure in x - t space, known as spatiotemporal orientation. The angle of tilt corresponds to the velocity of motion.

The motion energy model detects motion by detecting spatiotemporal orientation. It assumes a filter with tilted receptive fields that are oriented in x - t space, which are consistent with the

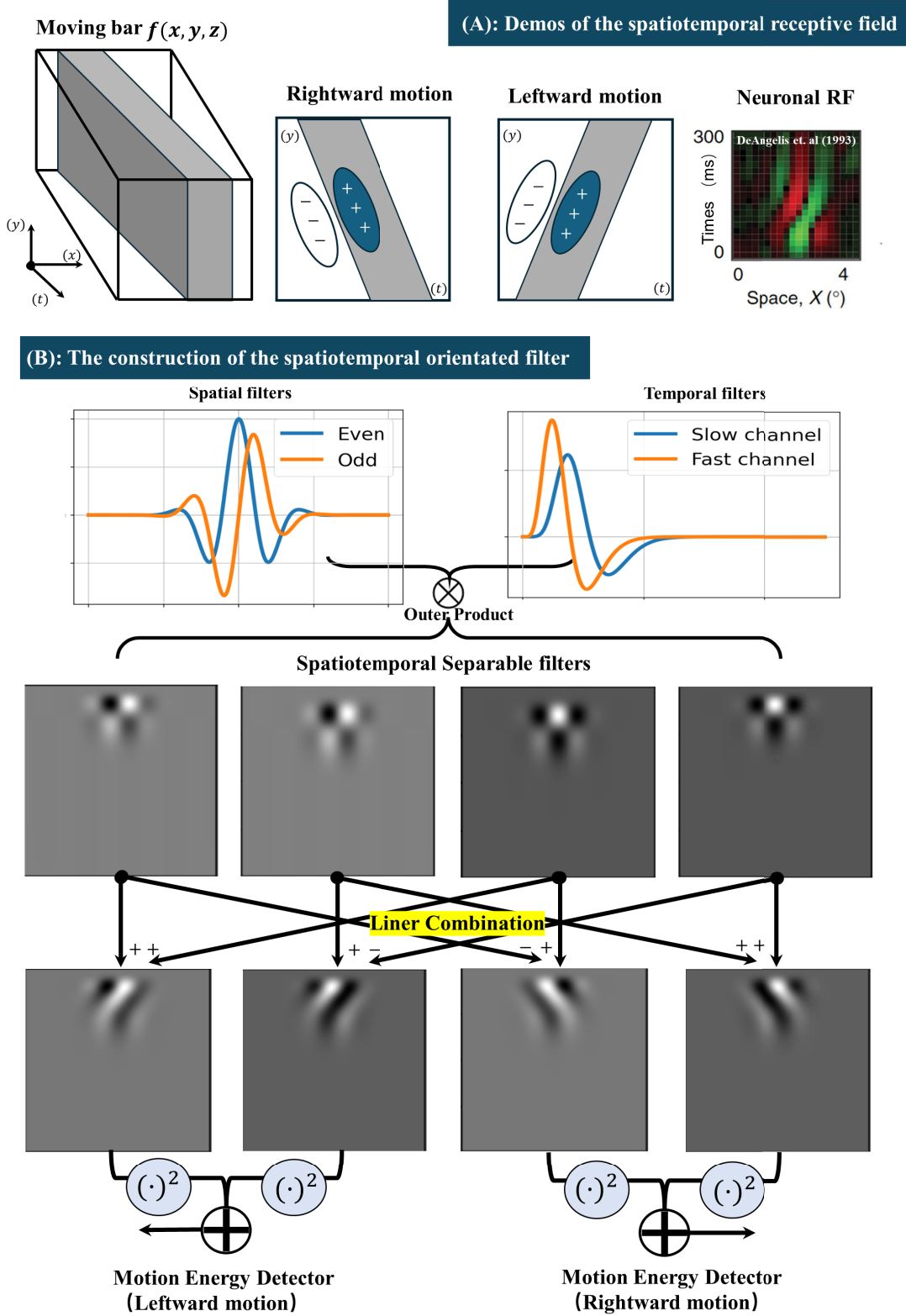


Figure 2.4: Motion energy model [2]. The neuronal receptive field is redrawn from [33].

previous neural records of neurophysiological studies [34].

The model has an elegant way of constructing these oriented receptive fields. Specifically,

spatiotemporally oriented receptive fields are created by two pairs of approximately orthogonal temporal and spatial filters. Two of these spatial filters are even functions, while the other two are odd functions. Considering that the Gabor function provides an accurate description of the spatial receptive fields of cortical cells [145], the section employs the Gabor function for the simulation, which is a sinusoidal function modulated by a Gaussian window. The even and odd variants are created using sine and cosine functions, respectively, defined by:

$$\mathcal{G}(x) = \exp\left(-\frac{x^2}{\sigma}\right); \begin{cases} \text{even}_x = \cos(2\pi s_f x) \cdot \mathcal{G}(x) \\ \text{odd}_x = \sin(2\pi s_f x) \cdot \mathcal{G}(x) \end{cases}$$

The temporal filters are biphasic, meaning they have both positive and negative response phases. Two variants of the temporal filters are used: one with a relatively fast response (Fast channel) and another with a relatively slow response (Slow channel). These are modeled using gamma functions with different parameters $\{n_{\text{slow}}, n_{\text{fast}}\}$, corresponding to the temporal impulse responses of mammalian visual systems, as shown in the first row of Figure 2.4 (B), corresponds to the following equation:

$$\begin{cases} \text{Slow}_t = (k \cdot t)^{n_{\text{slow}}} \cdot \exp(-k \cdot t) \left(\frac{1}{n_{\text{slow}}!} - \beta \frac{(k \cdot t)^2}{(n_{\text{slow}}+2)!} \right) \\ \text{Fast}_t = (k \cdot t)^{n_{\text{fast}}} \cdot \exp(-k \cdot t) \left(\frac{1}{n_{\text{fast}}!} - \beta \frac{(k \cdot t)^2}{(n_{\text{fast}}+2)!} \right) \end{cases}$$

By combining different temporal and spatial filters, a set of unoriented spatiotemporal filters can be constructed. Taking all potential combinations of even vs. odd spatial filters and fast vs. slow temporal filters, unoriented spatiotemporal filters are obtained through the outer product of temporal and spatial filters, which means they are spatially and temporally separable, illustrated by Figure 2.4 (B).

These unoriented filters can be linearly combined to produce a new set of spatiotemporally oriented filters, closely resembling the receptive fields required in subfigure A. There are two left-tuned filters and two right-tuned filters, with each pair being approximately orthogonal (due to the odd/even shift, with one offset by a quarter cycle relative to the other).

Finally, to ignore the positive and negative phase effects of motion, the squared sum of the convolution outputs of a pair of orthogonal spatiotemporal filters is performed, which produces a local motion detector selective for a specific direction (left or right). This design captures the activation pattern of direction-selective neurons well in the V1 area. Note that in subsequent steps, the full motion energy model includes energy normalization and the difference in energy between the left and right directions.

The motion models described above (spatiotemporal gradient model, Fourier analysis method, motion energy model), along with the Richard detector mentioned in chapter one, despite belonging to different approaches, share many common points. Firstly, they are all based on low-level processing of brightness shifts. Secondly, they share a fundamental assumption: brightness constancy across time. Although these models originate from different theories and experimental analyses, they all use brightness information to detect motion, and at certain stages, they are

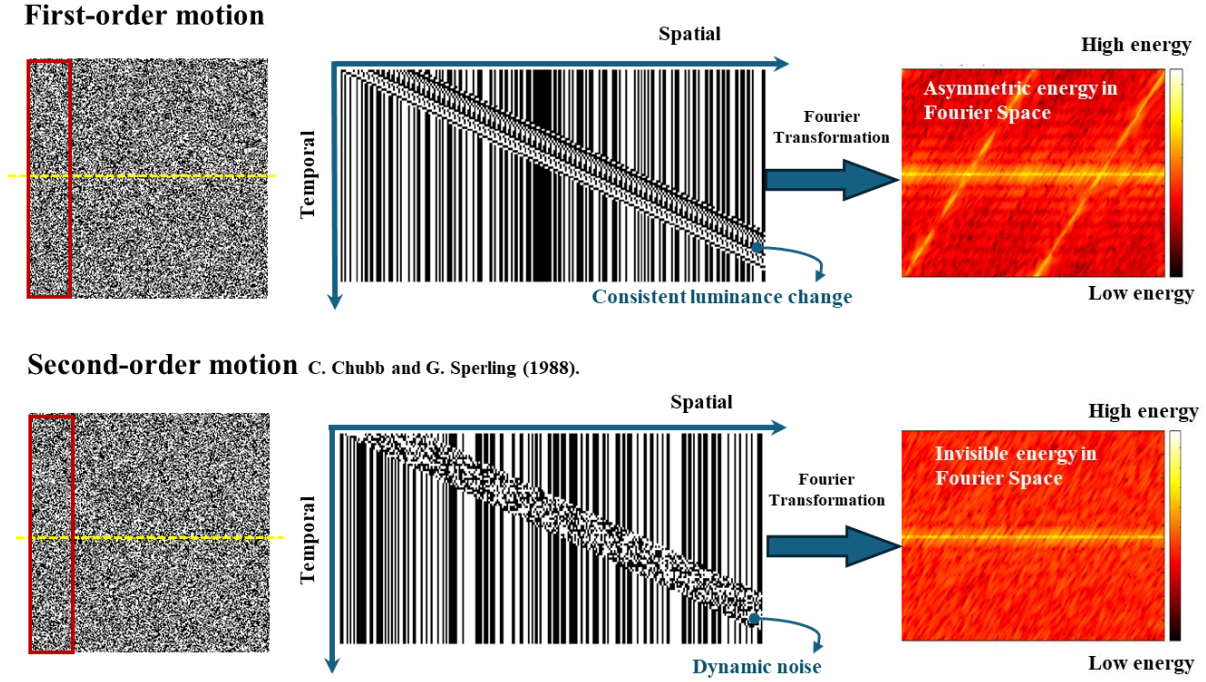


Figure 2.5: Illustration of first and second order motion (Drift-balanced motion) in spatiotemporal and Fourier space.

mathematically equivalent [2, 13]. Therefore, they are collectively referred to as first-order motion detection models or methods based on **Motion From Fourier Component (MFFC)** principles.

2.3.2 Second-order motion

Psychophysical experiments found that certain stimuli could induce a strong perception of motion that could not be explained by the Reichardt model or its equivalents. In 1970, Henning, Hertz, & Broadbent [59] first introduced the contrast-modulated patterns, noting that its detection could not be accomplished through linear filter systems. Sperling [157] was the first to mention the prototype of drift balance, then known as "uncorrelated motion." This complex yet biologically significant stimulus, which lacks a unique traceable pattern in both time and space, can still be clearly perceived by humans. Subsequently, such stimuli have been widely studied in visual perception.

The various motion phenomena discussed typically involve attributes such as spatiotemporal contrast, spatial density, or flicker frequency, rather than conventional luminance information. These characteristics are invisible in the frequency domain that first-order motion, detectable in the Fourier domain, operates within. Hence, these phenomena are referred to as **second-order motion**, or non-Fourier motion, as illustrated in Figure 2.5. This distinction is crucial because second-order motion stimuli, which cannot be accurately perceived by first-order motion models,

are essential for studying the underlying principles of second-order motion perception.

Following the detailed clarification by Chubb & Sperling in 1988 [25], research in visual motion perception expanded from focusing solely on luminance-based motion to encompassing motions defined by a wide range of non-luminance attributes. This shift ignited interdisciplinary research into these motion types. Over the years, many studies across various fields have provided new evidence to support and expand theoretical frameworks. However, debates continue regarding whether second-order motion is processed by a distinct mechanism. For instance, Johnston et al. [73, 74] used modeling techniques to demonstrate that some second-order motions could be detected by the same mechanisms that detect first-order motion. Nevertheless, overwhelming evidence from physiological and psychological experiments suggests that these two types of motion are encoded separately, at least initially, by distinct visual mechanisms [9, 10, 29].

Support for different mechanisms processing first-order and second-order motion stimuli arises from diverse fields, including psychophysics, neurophysiology, brain injury studies, and brain imaging. Mather et al. conducted psychophysical investigations [101–103] proposing a series of experiments on second-order motion, supporting the theory of separate mechanisms. Lucia et al. [177] have found patients with brain injuries who exhibit selective deficits in perceiving first-order motion but retain full perception of second-order motion. Experiments by Nishida et al. revealed different spatial frequency selectivities for first- and second-order motions [120, 122, 124, 125]. Electrophysiological studies by Baker et al. showed that in monkeys and cats, cells in certain brain areas respond differently to these two types of motion, with nearly half of the neurons in specific areas of the cat’s brain responding to contrast envelope stimuli [9, 10].

As physiological techniques continue to evolve, the neurobiological foundations of second-order motion perception in primates and humans have become increasingly supported. Current biological findings indicate that second-order motion is processed by mechanisms similar to, but distinct from, those used for first-order motion, at least initially. The widely accepted motion processing system includes at least two channels: a linear channel and a nonlinear channel. Many two-channel models involve early-stage filtering combined with nonlinear operations to extract second-order motion. Chubb, Sutter, et al. [24, 170] and Solomon, Sperling, et al. [156, 158] proposed a preprocessing filter comprising a low-pass temporal filter paired with another temporal derivative filter. Wilson et al. [185] suggested a preprocessing procedure that includes a directional spatial filter followed by a nonlinear squaring operation. The calculation of motion energy in the second-order channel uses a lower spatial frequency filter, employing filters of various orientations during the preprocessing stage to detect motion at texture boundaries. Similar two-stage models, combining fine-scale filters with subsequent coarse-scale filtering and rectification, are believed to be characteristic of cortical cells responding to contrast envelopes. Later, Smith [154] and Sperling [96] introduced a model that added an additional feature extraction process to the general first-order and second-order channels. Baker’s model employs a filter-rectify-filter approach to process second-order stimuli [199].

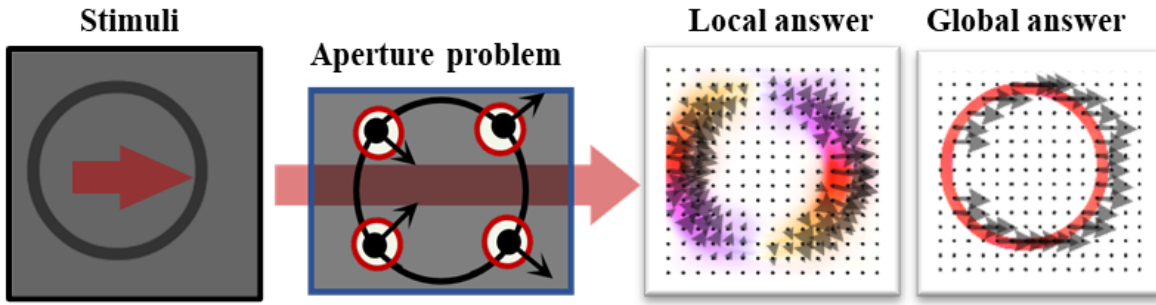


Figure 2.6: Aperture problem.

2.3.3 Motion Integration

A key goal of early visual motion processing is to estimate 2D motion vectors despite the ambiguity of local motion signals. However, the small receptive fields of directional neurons in V1 limit each to capturing only a part of a moving object. Due to their orientation selectivity, these neurons detect only motion perpendicular to an object's edge, leaving parallel components undetected. This limitation, known as the aperture problem, illustrates how each neuron's receptive field (RF) offers a restricted view of motion, making it challenging to fully specify the true 2D motion vector from 1D motion signals [43].

The previously discussed motion detectors, such as the Reichardt detector, motion energy model, and spatiotemporal gradients, are able to simulate V1's functionality at detecting local luminance shifts. However, these detectors are still in the initial stage of local motion detection, which is insufficient for accurate global integration, as demonstrated in Figure 2.6. The local motion detectors have limited receptive fields, perceiving motion only within a narrow scope, whereas accurately determining the physical moving direction requires a global perspective. To achieve accurate object motion estimation, the biological vision system possesses two key abilities: (1) detecting local motion and (2) having a global view of local motion and integrating the local motion adaptively.

In previous studies of visual motion perception, component stimuli (sinusoidal gratings with a single frequency component and moving direction) and plaid stimuli (consisting of two components with different parameters) have proven effective for examining this integration both perceptually and physiologically, as demonstrated in Figure 2.7 (A).

Psychophysical studies have shown that humans can integrate the motions of plaid components under various conditions. Theoretical methods for combining these vectors include the intersection of constraints (IoC), which is accurate but complex to implement physiologically [116], and a weighted vector-average (VA), which approximates correct motion effectively in most situations. The latter method's predictions often align with errors observed in visual processing during brief views or when the image is weakened by low contrast or noise [160]. The computational methods IoC and VA are illustrated on the right side of Figure 2.7 (A).

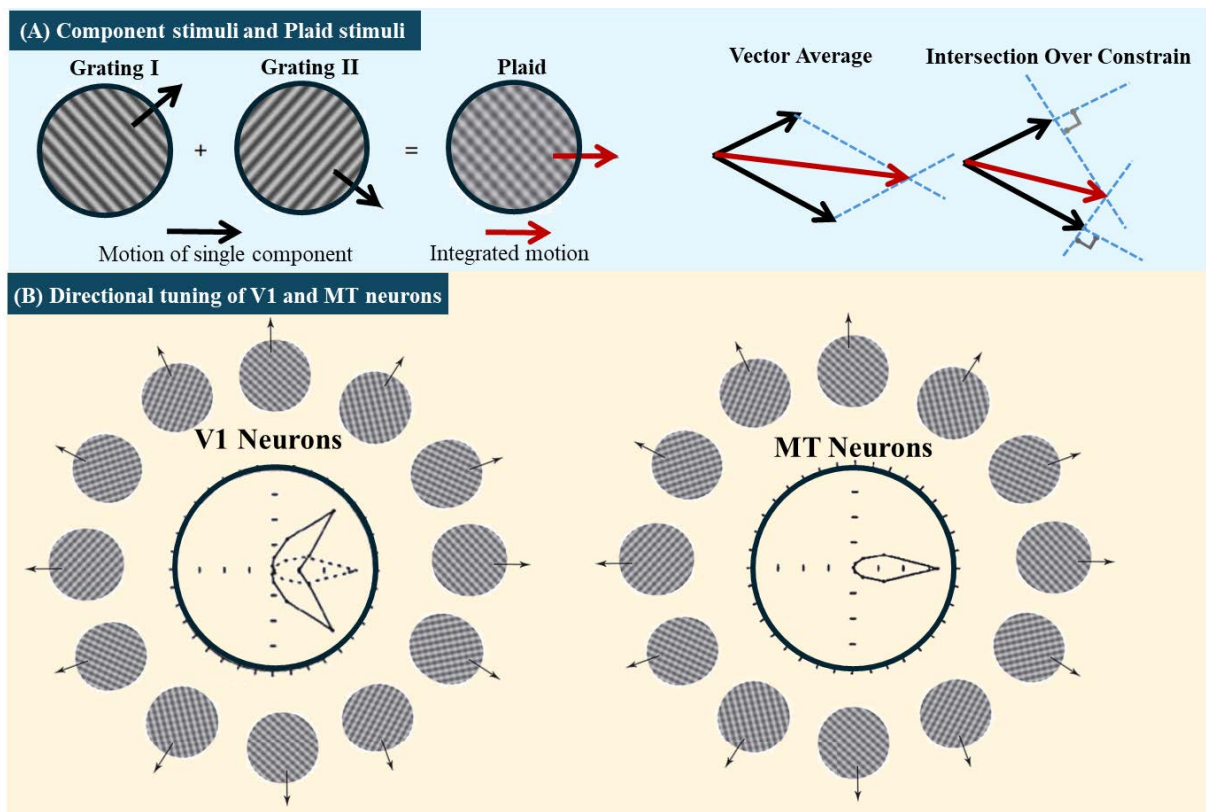


Figure 2.7: Motion integration using grating and paid. (A) Component stimuli and Plaid stimuli used the motion integration; (B) Directional tuning curve of V1 and MT neurons. Figure B is modified from [107].

However, Amano, Edwards, Badcock, and Nishida (2009) [4] demonstrated that the human visual system adaptively switches between two types of motion pooling—1D and 2D—based on the stimulus: In 1D motion pooling, local 1D motion signals are integrated simultaneously over orientation and space, suitable for stimuli with a single orientation (e.g., lines, Gabors) where the aperture problem necessitates spatial pooling. This process aligns with the Intersection of Constraints (IOC) rule, though a non-rigid interpretation might be employed when plausible. In contrast, 2D motion pooling occurs when stimuli have multiple orientations (e.g., dots, Gabor plaids), allowing local 2D motion vectors to be determined through cross-orientation integration and then spatially integrated. This method follows the Vector Average (VA) rule. The interplay between IOC and VA mechanisms is also evident in standard plaid perception, indicating a flexible, adaptive approach to motion integration in the visual system.

Physiologically, there is a general functional division where the striate visual cortex (V1) primarily serves local motion detection, with neurons sensitive to component stimuli. In contrast, the MT and MST regions are mainly responsible for higher-level functions such as motion integration and segregation. The directional tuning ability of neurons in response to plaids efficiently evaluates their motion integration capacity. Certain neurons in the MT area integrate multiple motion directions from plaid stimuli. When presented with a plaid moving in different directions, these neurons respond maximally to their preferred pattern direction, even if no single plaid component moves in that direction, as shown in Figure 2.7 (B).

Earlier models of motion integration in MT, like that proposed by Simoncelli and Heeger [152], incorporated a specific pattern of projection from V1 to MT with a slight nonlinearity in MT, effectively accounting for pattern direction selectivity and other phenomena. More recent elaborations of this model have been directly tested, revealing subtle variations in connectivity within MT that explain the range of responses from component to pattern direction selectivity. Although several computational models of area MT have been proposed [152, 184], a more comprehensive computational architecture that replicates the motion integration abilities of MT neurons and human perception, especially the human perception in complex natural scenarios, remains elusive [129].

2.4 Motion Estimation based on DNNs

In recent years, the rapid advancement of deep learning technology has significantly impacted motion estimation fields. Data-driven methods based on DNNs have garnered considerable attention due to their powerful representation capabilities and the availability of large datasets. These methods have achieved extremely high accuracy in several motion estimation benchmarks within the computer vision field, demonstrating human-matched capacity to estimate object motion in challenging environments. This progress highlights a new potential avenue of data-driven methods to develop human-aligned motion processing models. This section introduces several common optical flow datasets, followed by an overview of representative DNN models

and unsupervised learning methods.

2.4.1 Common Optical Flow Datasets

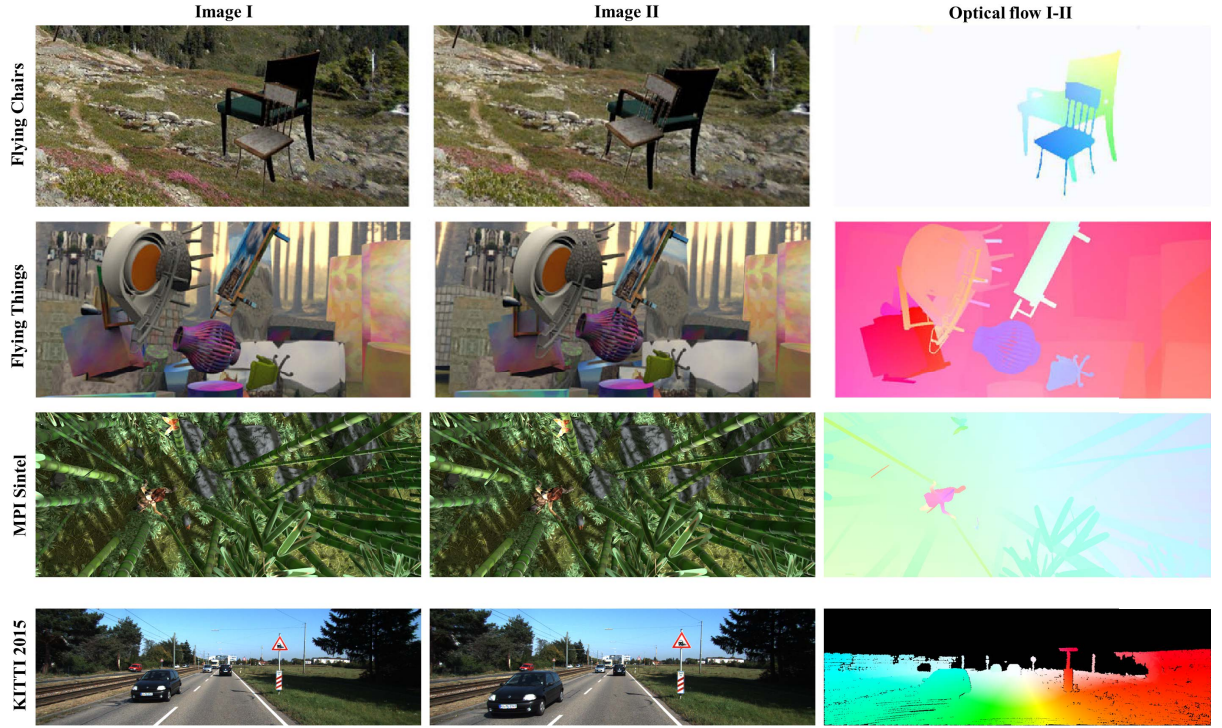


Figure 2.8: Demonstration of common optical flow estimation datasets. Adopted from Sintel [18], KITTI [114], Flyingthings [105], and Flying Chairs [39]

Flying Chairs [39]

The “Flying Chairs” dataset is a synthetic dataset with optical flow ground truth, consisting of 22,872 image pairs and their corresponding optical flow fields. The images depict 3D chair models moving in front of random backgrounds sourced from Flickr. The motions of both the chairs and the background are purely planar. Due to the dataset’s simplicity, featuring only rigid chairs and relatively simple scenes, it is commonly used as a pre-training dataset to initialize models, thereby enhancing their generalization capabilities.

FlyingThings3D [105]

The FlyingThings3D dataset is a synthetic dataset designed for optical flow, disparity, and scene flow estimation. It consists of everyday objects flying along randomized 3D trajectories, generating approximately 25,000 stereo frames with corresponding ground truth data. Rather than focusing on a specific task or enforcing strict naturalism, the dataset leverages randomness and a large pool of rendering assets to produce a vast amount of diverse data. This approach minimizes the risk of repetition or saturation, making FlyingThings3D a popular benchmark for training and evaluating various computer vision models.

MPI Sintel [105]

The MPI (Max Planck Institute) Sintel dataset is a prominent resource for evaluating optical flow, depth estimation, and image segmentation algorithms in computer vision. Derived from the open-source 3D animated short film Sintel, this dataset offers a combination of synthesized stereo images and ground truth data, making it a common benchmark and training set in current computer vision studies. The Sintel dataset comprises 1,064 stereo image pairs and their corresponding ground truth for optical flow, depth, segmentation, occlusion, as well as disparity maps. Each image in these pairs is rendered in RGB with a resolution of 1024×436 pixels and 8-bit per channel, while the optical flow maps are grayscale with the same resolution with 32 floats.

One of the key strengths of the Sintel dataset is its variety, as it includes 23 different scenes from the film. These scenes cover a wide range of environmental conditions, motion patterns, and complexities, providing a robust testbed for evaluating the performance and generalizability of optical flow and depth estimation algorithms. The scenes are presented in two versions: the “Clean Pass,” which features images without motion blur and noise, offering sharp and clear visuals; and the “Final Pass,” which includes realistic motion blur, atmospheric effects, and noise, mimicking real-world conditions more closely.

KITTI [113]

The KITTI dataset, developed by the Karlsruhe Institute of Technology and Toyota Technological Institute, is one of the most widely used datasets in mobile robotics and autonomous driving research. It comprises extensive traffic scenarios recorded using various sensor modalities, including high-resolution RGB and grayscale stereo cameras, as well as a 3D laser scanner.

The dataset encompasses multiple urban scene categories such as road, vertical, sky, building, vegetation, sidewalk, car, pedestrian, cyclist, sign/pole, and fence. Ros et al. further expanded the annotations to include classes like building, tree, car, sign, pedestrian, fence, pole, sidewalk, and bicyclist. KITTI provides a wide range of data formats and labels, including 1382×512 resolution stereo grayscale images, RGB images, point clouds, optical flow, 3D scene flow, depth, disparity, and camera parameters. For optical flow tasks, the KITTI Scene Flow dataset includes 200 training scenes and 200 test scenes. Each scene comprises 20 consecutive frames, with sparse optical flow annotations available for the 10th frame.

These comprehensive data and annotations format make the KITTI dataset popular for studies in autonomous driving and 3D computer visions.

2.4.2 Model Development

Existing deep neural networks for optical flow estimation can be broadly categorized into several types: encoder-decoder-based networks, spatial pyramid networks, iterative refinement structures based on recurrent networks, and transformer-based frameworks.

In 2015, Dosovitskiy et al. [38] introduced optical flow estimation methods using convo-

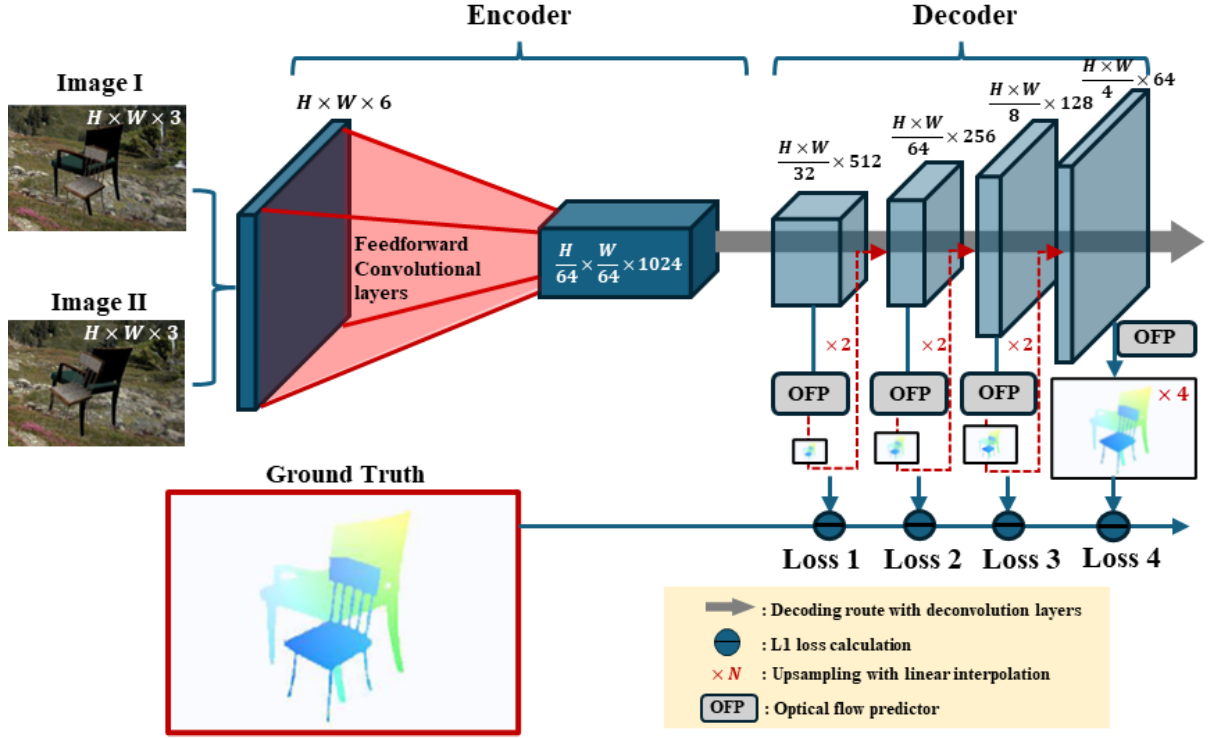


Figure 2.9: Demonstration of the encoder-decoder architecture of FlowNet [38]

lutional neural networks (CNNs). They designed two foundational networks, FlowNetS and FlowNetC, which follow an encoder-decoder architecture. These networks comprise two main parts: the encoding part and the decoding part, as depicted in Figure 2.9.

Both architectures utilize a series of convolutional layers in the encoding part, with the inputs being two consecutive frames (Image 1 and Image 2). These images are processed through multiple convolutional layers to extract correspondence features. Due to a stride setting of 2, the resolution of the feature maps decreases as the network deepens. FlowNetS and FlowNetC differ slightly in their encoding parts. FlowNetS uses a single input path where the two images are stacked and fed into the network to extract combined features. In contrast, FlowNetC uses two separate input paths for the two images, extracting features independently before merging them using a correlation layer to calculate the matching cost, which is then fed into the decoding part.

The decoding parts of FlowNetS and FlowNetC are identical, as illustrated on the right side of Figure 2.9. They consist of a series of deconvolution layers with a stride of 2, gradually expanding the resolution of the feature maps. Generally, there are five decoding stages with scales increasing across $\{\frac{1}{32}, \frac{1}{16}, \frac{1}{8}, \frac{1}{4}, 1\}$, with each scale generating a refined optical flow for the next scale.

The networks are trained end-to-end in a supervised manner, requiring ground truth labels during training. The loss function can utilize general distance metrics between flow vectors, such as MAE (Mean Absolute Error), MSE (Mean Squared Error), and EPE (End-Point Error),

between the predicted flow and ground truth flow, with a summation across all scales. EPE is also a common method for evaluating model performance, which calculates the Euclidean distance between the endpoints of the predicted $y = (u, v)$ and ground truth $y^* = (u^*, v^*)$ motion vectors, as defined in Equation 2.3.

$$\text{EPE}(y, y^*) = \sqrt{(u - u^*)^2 + (v - v^*)^2} \quad (2.3)$$

As the first work to incorporate learning into optical flow tasks, FlowNet achieved performance comparable to traditional methods while significantly reducing runtime, leading to many subsequent improvements based on its architecture.

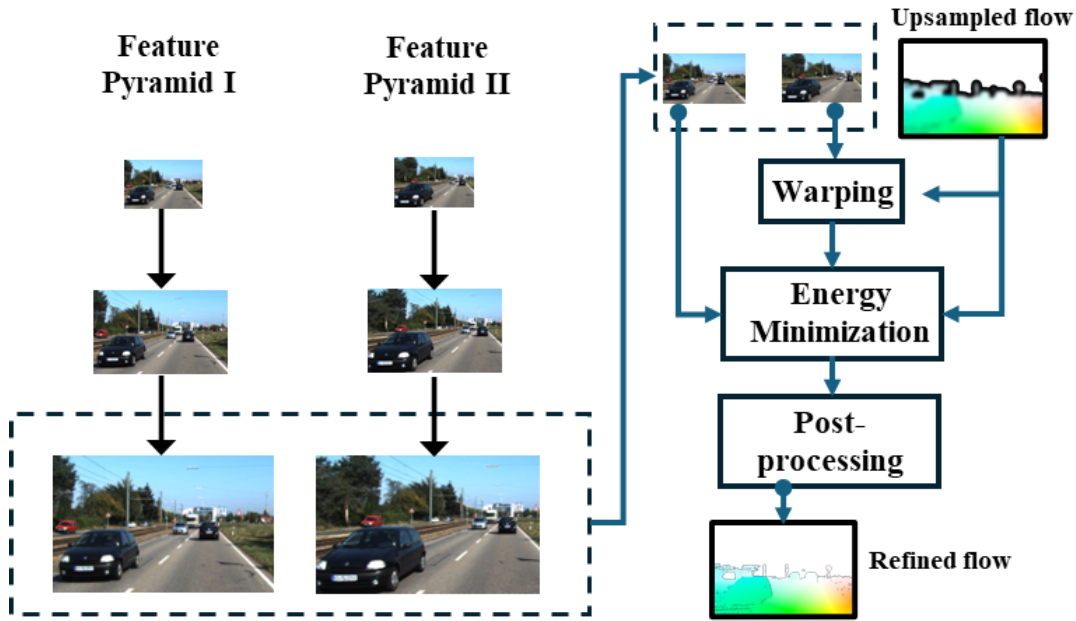


Figure 2.10: The architecture of the spatial pyramid network (SpyNet [139]).

In 2017, Ranjan et al. [140] proposed using a spatial pyramid network for optical flow estimation. This network incorporates a coarse-to-fine strategy into the flow estimation process, enhancing computational efficiency. As shown in Figure 2.10, the architecture starts with defining a pyramid framework. At each pyramid level, a 2D geometric warp operation is used to match two images based on the optical flow from the previous pyramid level. An optical flow estimator then calculates the matching cost and refines the current flow field, which is then upsampled for image warping and corresponding matching at the next level. This core strategy calculates small displacements (incremental flow) at each level from coarse to fine scales, rather than directly estimating large displacements, thereby reducing computational requirements and achieving results comparable to FlowNet on public datasets like MPI-Sintel and KITTI while significantly reducing model size.

Following the foundational architecture proposed in the literature, several improved supervised learning networks have been developed. Sun et al. [164] introduced correlation layers at

each pyramid level and replaced image warping with feature map warping to minimize spatial distance between features of adjacent frames. The architecture, illustrated in Figure 2.11, consists of a feature pyramid, an optical flow estimator, and a post-processing sub-network. Input images pass through a series of convolutional layers to extract multi-scale features. The optical flow estimator, a fully convolutional network, estimates flow at each pyramid level, while the post-processing network refines the estimated flow. Each level's estimated flow is compared with the ground truth to calculate EPE loss, enhancing precision while maintaining a lightweight model. Such a strategy can also be modified as a recurrent protocol [66], largely increasing parameter efficiency.

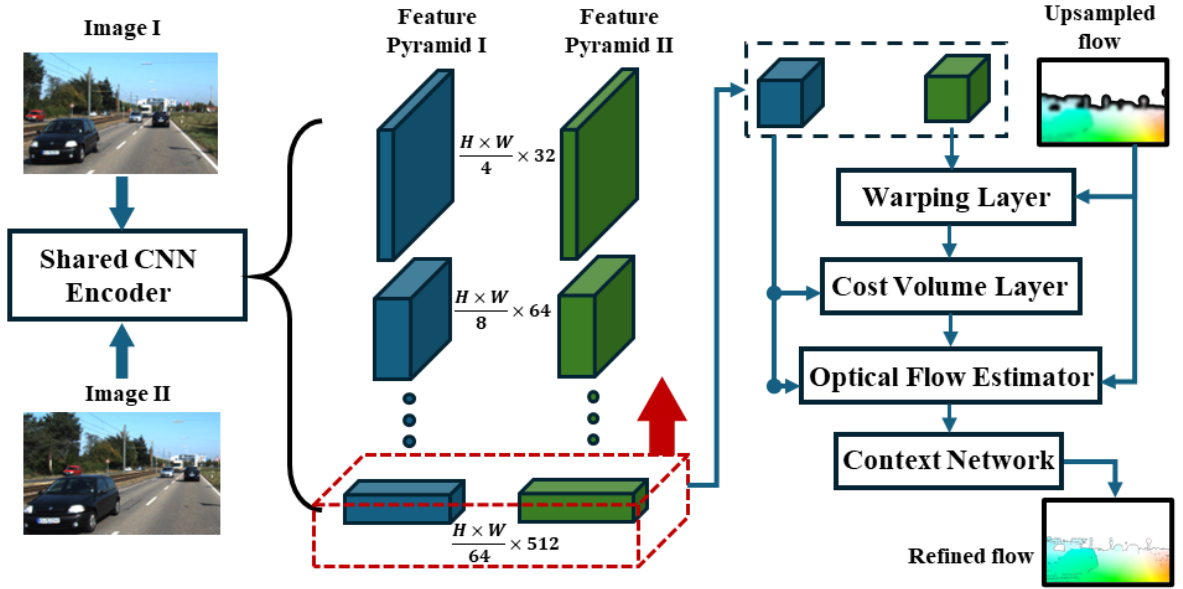


Figure 2.11: Pyramid network for optical flow estimation (PWCNet [164])

In 2020, Zachary Teed and Jia Deng [171] introduced Recurrent All-Pairs Field Transforms (RAFT), a novel deep network architecture for optical flow estimation. RAFT retains feature warping and matching cost calculation but eliminates the multi-scale feature pyramid. Instead, it extracts features at a uniform $1/8$ scale and constructs multi-scale 4D correlation volumes for all pixel pairs. In the decoding phase, RAFT incorporates a recurrent neural network to iteratively update the flow field, optimizing the flow in a coarse-to-fine manner by performing lookups on the correlation volumes.

RAFT achieves state-of-the-art performance, with an F1-all error of 5.10% on the KITTI dataset, a 16% reduction from the previous best result. On the Sintel dataset, RAFT also performs exceptionally well, demonstrating strong cross-dataset generalization and high efficiency in inference time, training speed, and parameter count.

Recently, with the development of the attention mechanism [178] and Transformers [37], several studies have started to utilize the transformer architecture in the optical flow estimation task to replace the CNNs [62, 187]. The attention mechanism from Transformer is powerful in

establishing the flexible connection that convolution operations and it's also reasonable to extract the correspondence relationship by calculating the cross attention between pairs of frames over time.

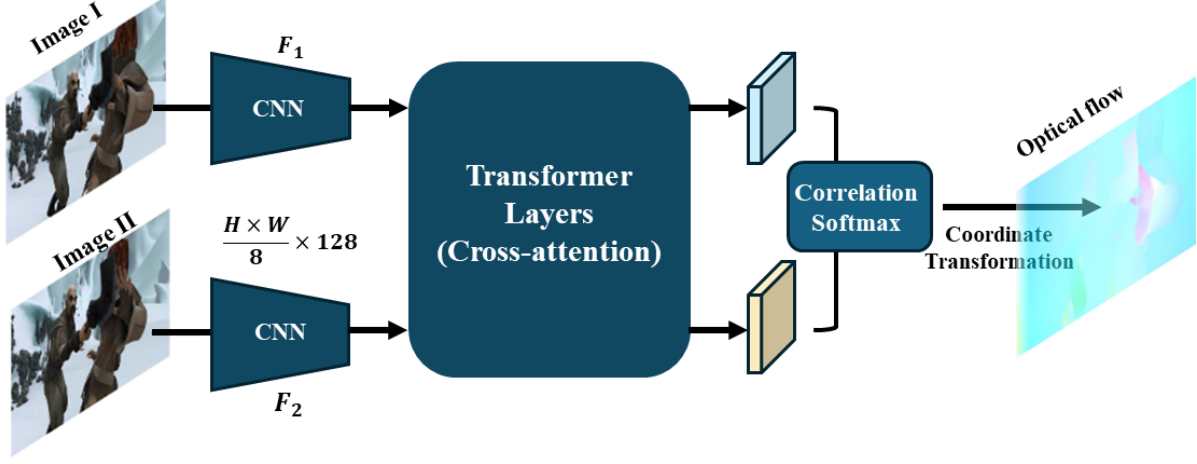


Figure 2.12: Demonstration of GMFlow [187].

The GMFlow method [187] is one of the representative approaches that incorporate the Transformer architecture into the optical flow estimation task. As illustrated in Figure 2.12, given two consecutive video frames I_1 and I_2 , GMFlow first extracts downsampled dense features $F_1, F_2 \in \mathbb{R}^{H \times W \times D}$ using a weight-sharing convolutional network, where H , W , and D denote height, width, and feature dimension, respectively. Considering that correspondences in the two frames should share high similarity, the feature similarity for each pixel in F_1 is compared with all pixels in F_2 by computing their correlations. This can be implemented efficiently with a simple matrix multiplication:

$$C = \frac{F_1 F_2^T}{\sqrt{D}} \in \mathbb{R}^{H \times W \times H \times W},$$

where each element in the correlation matrix C represents the correlation value between coordinates $p_1 = (i, j)$ in F_1 and $p_2 = (k, l)$ in F_2 , and $\frac{1}{\sqrt{D}}$ is a normalization factor to avoid large values after the dot-product operation [178].

To identify the correspondence with end-to-end training, GMFlow uses a differentiable matching layer [79, 186]. Specifically, the last two dimensions of C are normalized with the softmax operation, resulting in a matching distribution:

$$M = \text{softmax}(C) \in \mathbb{R}^{H \times W \times H \times W}.$$

For each location in F_1 with respect to all locations in F_2 , the correspondence $\hat{G}$ can be obtained by taking a weighted average of the 2D coordinates of the pixel grid $G \in \mathbb{R}^{H \times W \times 2}$ with the matching distribution M :

$$\hat{G} = MG \in \mathbb{R}^{H \times W \times 2}.$$

Finally, the optical flow $\mathbf{V}$ can be obtained by computing the difference between the corresponding pixel coordinates:

$$\mathbf{V} = \hat{\mathbf{G}} - \mathbf{G} \in \mathbb{R}^{H \times W \times 2}.$$

This softmax-based approach not only enables end-to-end training but also provides sub-pixel accuracy.

2.4.3 Unsupervised Learning Methods

Unsupervised learning frameworks are proposed to utilize the resources of unlabeled videos [193], first introduced by Yu et al. [193], which built the loss function based on brightness constancy and motion smoothness, similar to conventional methods' energy minimization. Multiple subsequent studies have refined the optimization objective, including occlusion reasoning [110, 182], multi-frame extension [53, 71], epipolar constraint [198], and stereo depth [86, 181].

Although there are many other strategies [69, 87, 90, 159] to solve the unsupervised training problem, we only consider the fundamental method here, which is based on image warping and consistent matching strategies. The overall strategy behind this is adopting a photometric loss that measures the difference between the target image and the (inversely) warped source image based on the dense flow field predicted from the network.

Let a pair of consecutive images $\mathcal{I} = \{I_a, I_b\}$, the target is to estimate current optical flow $F_{a \rightarrow b} \in \mathbb{R}^{H \times W \times 2}$ from $\mathcal{I}$, i.e. , $F_{a \rightarrow b} = f(\mathcal{I}; \Theta)$, where Θ is a set trainable parameters of the model. We follow the general warping-based strategy as our basic unsupervised learning approach, by which the network can be trained implicitly with view synthesis. Specifically, each image I_a could be reconstructed by its next frame I_b via inverse warping operation, i.e., $\hat{I}_a = I_b(\mathbf{p} + F_{a \rightarrow b}(\mathbf{p}))$, where $\mathbf{p} \in \mathbb{R}^{H \times W}$ is the spatial location across whole images. Utilizing bilinear interpolation, the method can be realized in a differentiable way at the sub-pixel level. Then one only needs to define the consistency evaluation function $\rho(\cdot)$ between the reconstructed image $\hat{I}_a$ and the original image I_a , i.e. $\mathcal{L}_{\text{ph}} \sim \sum_{\mathbf{p}} \rho(\hat{I}_a(\Theta), I_a)$, so as to implicitly supervise the generated optical flow. The basic similarity evaluation loss is L1, or Charbonnier loss [163], which simply considers the pixel-wise similarity between a pair of images. For instance, the Charbonnier loss is defined as:

$$\mathcal{L}_{\text{charb}} = \sum_{\mathbf{p}} \rho(\hat{I}_a(\Theta), I_a) = \sum_{\mathbf{p}} \sqrt{(\hat{I}_a(\mathbf{p}) - I_a(\mathbf{p}))^2 + \epsilon^2} \quad (2.4)$$

where: ϵ is a small constant to ensure numerical stability when the difference between $\hat{I}_a(\mathbf{p})$ and $I_a(\mathbf{p})$ is very small.

The more robust loss that focuses on structural similarities, like SSIM loss and Census loss [110], are widely used in the community to overcome the variations in color and brightness across frames. The Structural Similarity Index (SSIM) loss is defined as follows:

$$\text{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}$$

where μ_x and μ_y are the mean intensities of x and y ; σ_x^2 and σ_y^2 are the variances of x and y ; σ_{xy} is the covariance of x and y ; C_1 and C_2 are constants to stabilize the division when the denominators are close to zero. The SSIM loss can be defined as:

$$\mathcal{L}_{\text{SSIM}} = \sum_{\mathbf{p}} \frac{1 - \text{SSIM}(\hat{I}_a(\mathbf{p}), I_a(\mathbf{p}))}{2}, \quad (2.5)$$

which evaluates the structural similarity between the reconstructed image $\hat{I}_t$ and the original image I_t .

The Census Transform is defined by comparing each pixel to its neighbors, creating a bit-string:

$$\text{Census}(I, \mathbf{p}) = \prod_{\mathbf{q} \in \mathcal{N}(\mathbf{p})} \delta(I(\mathbf{p}), I(\mathbf{q})),$$

where $\mathcal{N}(\mathbf{p})$ is the set of neighboring pixels of $\mathbf{p}$ and $\delta(I(\mathbf{p}), I(\mathbf{q}))$ is a binary function that returns 1 if $I(\mathbf{p}) > I(\mathbf{q})$, and 0 otherwise. The Census loss can then be defined as the Hamming distance between the Census Transforms of $\hat{I}_a$ and I_a :

$$\mathcal{L}_{\text{census}} = \sum_{\mathbf{p}} \text{Hamming}(\text{Census}(\hat{I}_a, \mathbf{p}), \text{Census}(I_a, \mathbf{p})), \quad (2.6)$$

where $\text{Hamming}(\cdot, \cdot)$ calculates the Hamming distance between two bitstrings, which can be implemented in a differential way [110]. These robust loss functions help mitigate the effects of color and brightness variations, focusing more on the structural consistency between frames.

Due to the aperture problem and the ambiguity of local appearance, supervision solely based on the photometric loss does not sufficiently constrain the problem for somewhere textureless or with repetitive patterns. One of the most common ways to reduce ambiguity is named edge-aware first and second-order smooth regularization [174]. Given frame I_{t-1} with flow field $F_{t-1 \rightarrow t}$, the k -order edge-aware loss $\mathcal{L}_{sm(k)}$ could be defined as:

$$\begin{aligned} \mathcal{L}_{sm(k)} &\sim \sum_{\mathbf{p}} \left(w_{\mathbf{p}}^x \cdot \left| \frac{\partial^k F_{t-1 \rightarrow t}}{\partial x^k} \right|_{\mathbf{p}} + w_{\mathbf{p}}^y \cdot \left| \frac{\partial^k F_{t-1 \rightarrow t}}{\partial y^k} \right|_{\mathbf{p}} \right), \\ w_{\mathbf{p}}^x &= \exp \left(-\lambda \sum_c \left| \frac{\partial I_{t-1}}{\partial x} \right|_{\mathbf{p}} \right), \\ w_{\mathbf{p}}^y &= \exp \left(-\lambda \sum_c \left| \frac{\partial I_{t-1}}{\partial y} \right|_{\mathbf{p}} \right). \end{aligned} \quad (2.7)$$

Where $w_{\mathbf{p}}^x$, $w_{\mathbf{p}}^y$ is the attenuation weight determined by the smoothness of the first-order image gradient, which intuitively forces the flow field in areas with similar pixels to be smoother. Following previous work [77, 87], we use first-order smooth loss on Sintel dataset, second-order smooth loss on KITTI dataset.

Motion occlusion is a challenging problem in unsupervised learning, which results in pixel-wised information loss from I_t to I_{t-1} . The most common method is to circumvent the loss calculation for the occluded region by inferring the binary occlusion mask $O_{t-1 \rightarrow t} \in \{0, 1\}^{H \times W}$

from I_{t-1} to I_t , where $O_{t-1 \rightarrow t}(\mathbf{p}) = 0$ means the locations $\mathbf{p}$ from I_{t-1} are occluded in I_t and vice versa. Following the previous method [77, 87], a forward-backward check [70] is used in our basic approach to estimate the occlusion mask.

We simplify the notion in the following section. Given two adjacent time step $\{a, b \mid b > a\}$, one denotes forward flow $F_{a \rightarrow b}$ as F_a , and backward flow $F_{b \rightarrow a}$ as F_b^{-1} ; $\mathbf{p} \in \mathbb{R}^{H \times W}$ is denoted as spatial position set of a specific image; $\mathcal{W}_a(\cdot) : I_b \mapsto \hat{I}_a$ is denoted as warping operation from I_b via $F_{a \rightarrow b}$; $\mathbf{E}_{\mathbf{p}}(\cdot)$ represents the statistical expectations along the $\mathbf{p}$.

Chapter 3

Elaboration of Machine Vision: Human-motivated Unsupervised Learning of Motion Estimation

Humans and animals learn to estimate motion through spontaneous interaction with the world, conceptually resembling unsupervised learning, where motion rules are understood without explicit ground-truth labels. With the ultimate goal of constructing a more human-aligned model, this chapter incorporates techniques from computer vision (CV) to explore unsupervised motion estimation methods. We will integrate this unsupervised method with our human-aligned motion processing model proposed in Chapter 5.

Our exploration proceeds in two aspects. On the one hand, we aim to develop a solution that mimics the learning capability of humans to achieve accurate motion estimation without explicit ground truth. To construct a computable image method, we employ techniques based on warping reconstruction from the computer vision field. On the other hand, current CV methods face several challenges, particularly when pixel information is lost due to occlusions and motion blur or when pixel correspondence is impaired by variations in image content and spatial structure over time. These challenges result in the inability to reconstruct the source image by warping, leading to confusion about the lost object. Therefore, we aim to propose solutions to overcome these challenges in unsupervised learning methods.

Inspired by humans' consistent understanding/ estimation of motion in dynamic environments, we hypothesize that learning in multi-frame settings and memorizing motion history, similar to a temporal recurrent network, could enhance understanding of motion laws and improve performance in challenging situations. For instance, areas that lose explicit correspondence due to occlusions or motion blur might have been visible in previous frames, allowing for more reliable predictions based on temporal motion history, assuming the motion remains smooth under a sufficient sampling rate.

We integrate temporal dynamic modeling into the unsupervised learning of optical flow by

developing a lightweight temporal recurrent network. This network is designed to infer current motion by iteratively processing the causal image sequence and memorizing previous motion history to enhance prediction accuracy in subsequent frames. Using this model, we simulate various dynamic scenarios in unlabeled natural scenes, such as occlusion phenomena in multi-frame environments. We employ a self-supervised distillation method to provide more reliable pseudo-labels for these challenging scenes. This approach diverges from traditional two-frame methods that do not account for temporal dynamics and simply augment the scene discretely. We will demonstrate that this systematic simulation of dynamic scenes offers greater benefits compared to previous augmentation methods.

Moreover, while existing optical flow benchmarks like Sintel and Flying Chair provide realistic simulations, they rely on time-consuming 3D modeling and rendering engines. Instead, our focus is on incorporating augmentation on unlabeled real images in an online manner during training. Experiments on various optical flow benchmarks have demonstrated the effectiveness of our human-motivated approach, achieving performance comparable to several representative methods while offering advantages in memory and computational efficiency.

Further, various complex situations in natural environments, such as object occlusion, motion blur, limited perspective, and variations in lighting conditions, make accurate correspondence challenging. Addressing this issue, this study further improves the model architecture by decoupling the problem into two sub-tasks: one is to search for determined correspondence within a pair of frames, and the other is to manage mismatched regions due to occlusion, blur, non-Lambertian reflection, etc., by incorporating more spatial and temporal context information. To compensate for low-confidence regions, the proposed network adaptively identifies regions with correspondence confidence and utilizes temporal and spatial smoothness assumptions for motion re-prediction. Combined with previous simulations of scene variation, the proposed model can efficiently learn a variety of complex scenes in a multi-frame environment to optimize low-confidence regions. Experimental results demonstrate that the proposed model can run at high speed in real-time tasks while maintaining high accuracy, achieving state-of-the-art results on the Sintel Clean and Final benchmarks.

3.1 Problem Setting and Motivation

Unlike other vision tasks, where ground-truth labels for dense optical flow are rarely available for real image pairs, supervised optical flow techniques are typically trained using synthetic data. However, an inherent mismatch exists between real and synthetic data, leading to a disparity in data distribution that needs to be addressed to overcome this challenge [104]. The drawback of using over-intrinsic label rendering and the limitation of synthetic data enhances the viability of unsupervised optical flow estimation.

Unsupervised learning frameworks use unlabeled videos [193], deriving a photometric loss that measures the difference between the target image and the (inversely) warped source image

based on the dense flow field predicted by the network. However, image-warping-based unsupervised learning faces multiple challenges, including brightness and color variations and motion blur in multi-frame dynamic environments. More importantly, as the images are 2D projections of 3D space, spatial occlusion induced by objects in motion causes pixel information loss over frames that cannot be recovered by image warping. Such factors mislead the networks and degrade performance.

3.1.1 Problem I: Motion is a Dynamic Process

Several studies have sought to address these problems by designing more robust loss functions [69, 110], using masks to circumvent the supervision of occluded regions [182], and tracking occluded pixels in adjacent frames [71]. Based on self-supervised learning, another line of studies augments images by adding random noise patches to simulate occluded scenes [87, 90, 99]. However, a prevalent limitation observed is that most existing approaches for estimating optical flow use pairs of images. Although a few methods have employed multiple frames [71, 87], these still use temporally static feedforward CNNs without recurrent feedback in the time dimension. Moreover, motion is highly related to temporal dynamics, including factors like illumination shifts, spatial-perspective translations, and motion-induced phenomena such as blur and occlusion, underlining the significance of temporal dynamic relationships. Yet, existing self-supervised learning methods tend to apply arbitrary transformations based on double frames, such as randomly adding regional noise corruption to one image frame to simulate occlusion [87, 159].

3.1.2 Problem II: Motion is not only Correspondence Matching

In existing models, the most popular methods [66, 171] use a progressive strategy to search for correspondences across the feature map and gradually refine the optical flow. A critical step in this process is calculating the similarity at each location in the feature map, typically implemented by the inner product distance. For instance, PWCNet and RAFT calculate the correlation volume between features and feed it into the flow estimator. In contrast, GMFlow uses the inner product distance to match global correspondences and infer optical flow. However, in complex environments, it is challenging to resolve correspondences between two images. A typical example is that object occlusion during motion causes direct pixel loss. Additionally, local blurring of the image, such as motion blur, camera out-of-focus blur, and global or regional variations in lighting conditions, prevents accurate identification of these abnormal areas, causing outliers in the correlation volume. For simplicity, we refer to these regions as the **Low Matching Confidence** (LMC) regions.

3.1.3 Proposal I: Temporal Dynamic Modelling & Scene Augmentation

To address the aforementioned Problem I, we propose that motion perception should be learned in temporally dynamic environments to better grasp the essential laws of motion. We intro-

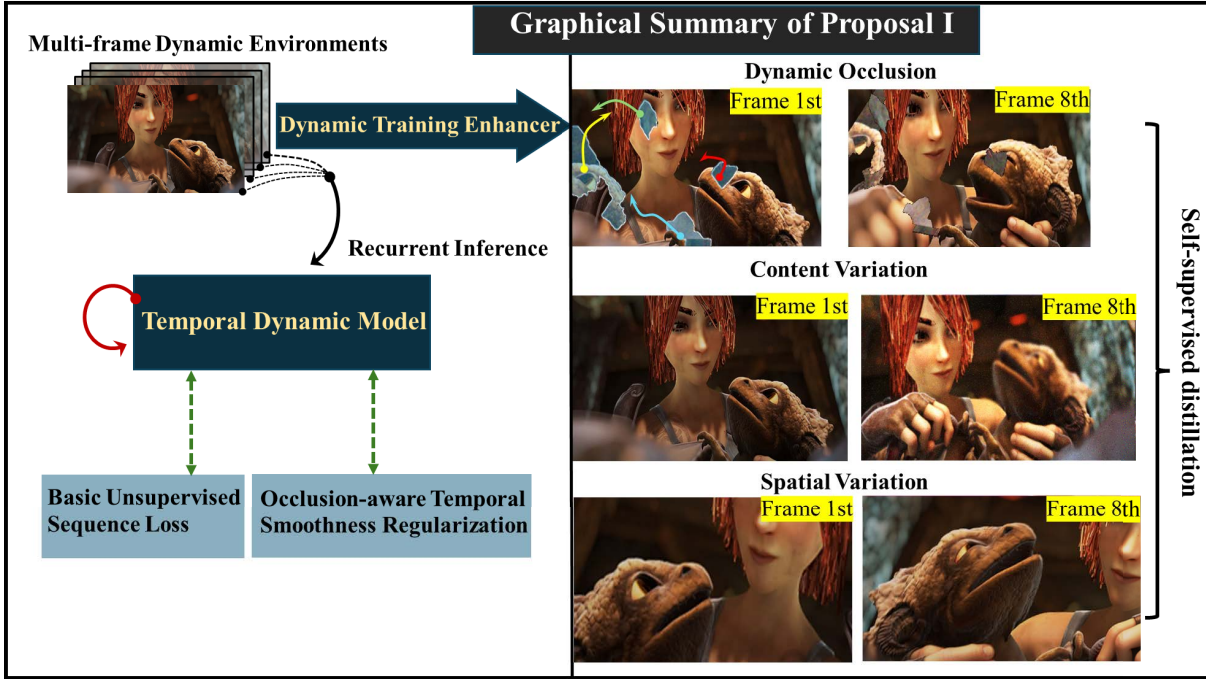


Figure 3.1: Graph summary of temporal dynamic scene augmentation.

duce an approach for estimating optical flow in dynamic environments, namely **ULDENet** (Unsupervised Learning in Dynamic Environments). This model is trained using temporal causal sequences to ensure a comprehensive understanding of object occlusion and scene dynamics. To achieve this, we incorporated three dynamic training enhancers (DTE) based on self-supervised learning: the **Dynamic Occlusion Enhancer** (DOE), **Content Variation Enhancer** (CVE), and **Spatial Variation Enhancer** (SVE).

The DOE extracts sub-object blocks from an original data distribution and simulates the continuous motion of objects across multiple frames, allowing the model to understand object occlusions over extended periods. We employ a mixed supervision strategy, combining unsupervised and self-supervised losses during DOE training, which provided a reliable pseudo label to occluded regions, thus enhancing generalization to occlusion scenes.

The CVE simulates variations in image attributes, such as object brightness and chromaticity dynamics over time. Additionally, it simulates regional motion blur and defocus blur caused by high-speed movement, thereby improving the model’s robustness in handling such challenging scenes. Lastly, the SVE simulates spatial object variations caused by camera shake and vibration in dynamic environments, as well as gradual translations and rotations of the viewpoint.

To evaluate the proposed augmentation strategy, we modified a lightweight PWCNet for our experiments, incorporating a spatiotemporal dual recurrent block. By leveraging the temporal smoothness of motion, these temporal dynamic networks propagate prior motions through hidden states, providing more reliable references for untraceable pixels. Initial experiments demonstrated that our method, with its few parameters and efficient inference speed, exhibits

stable convergence over sequences of variable lengths and outperforms counterparts in ablation studies.

3.1.4 Proposal II: A Decoupled Spatial-Temporal Adaptive Fusion Network

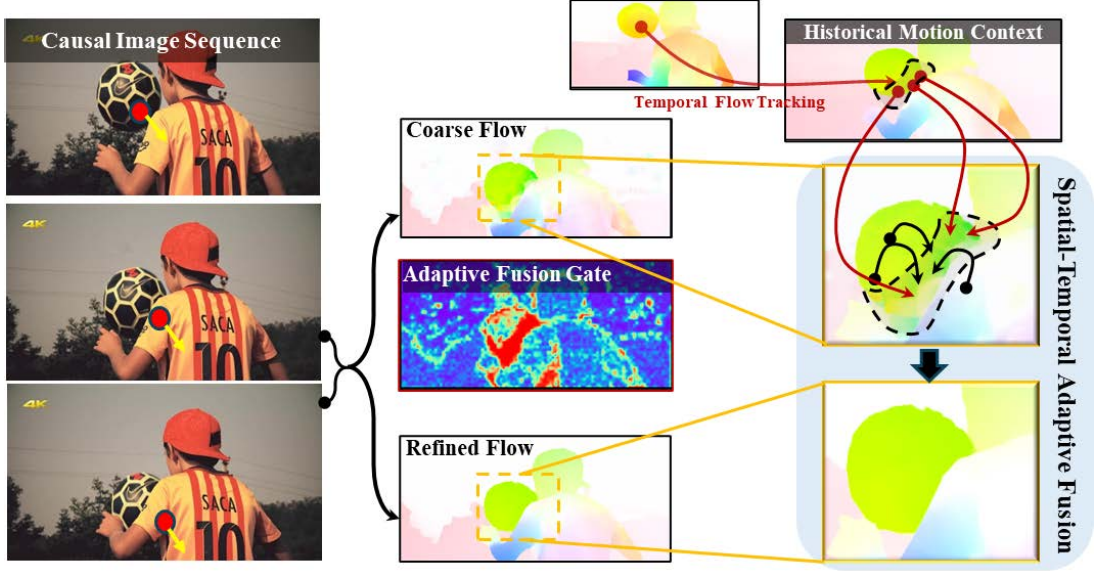


Figure 3.2: Demonstration of spatiotemporal adaptive fusion: No clear correspondence is available at the part region of the soccer ball due to occlusion, which is identified adaptively and re-predicted by tracking historical and spatial motion contextual information. Figure is adapted from [168].

Proposal II constructs a temporal dynamic model to perceive motion, consisting of two parts: **Global Correspondence Matching (GCM)** and **Spatio-Temporal Adaptive Fusion (STAF)** blocks. The GCM is a static model that finds the apparent correspondences between two images, while the STAF component is a temporal recursive model that adaptively identifies regions challenging to match and combines temporal and spatial contextual information to optimize low-confidence regions. Accordingly, the training method is decoupled into two parts to optimize both LMC and non-LMC areas in a targeted manner. Similar to how humans perceive motion, our model learns motion estimation in causal sequences, using only previous motion history information.

Due to Newton’s first law of motion, an object always has a continuous motion state within a short time window, meaning that objects occluded in the current frame are usually trackable in the previous one. Therefore, the historical motion states of the objects provide adequate temporal contextual information for current motion estimation, especially for the LMC regions. Building on the scene augmentation method in Proposal I, the STAF blocks are self-supervised and trained in a multi-frame environment, enabling the network to understand motion occlusion relations and use motion history to compensate for predicting existing LMC regions. This

also helps the network understand various types of LMC phenomena through self-supervised distillation.

To the best of our knowledge, by combining Proposal I and Proposal II, our approach achieves the best results among the few studies that have applied temporal dynamic models in unsupervised optical flow estimation tasks. Additionally, our model demonstrates state-of-the-art performance on the Sintel Clean and Final benchmarks.

3.2 Discussion with Related Works

Supervised methods require annotated flow ground truths to train networks. FlowNet [38] was the first work to propose learning optical flow estimation by training a fully convolutional network on the synthetic dataset, FlyingChairs. Then, FlowNet2 [68] proposed stacking multiple networks as an iterative improvement. SpyNet [139] is a spatial pyramid network that estimates optical flow in a coarse-to-fine manner and handles challenging scenes with large displacements. PWC-Net [164] and LiteFlowNet [65] warp features and calculate the cost volumes of all pyramid layers from the intermediate feature maps of CNNs. IRR-PWC [67] designs pyramid networks using iterative residual refinement schemes. Recently, RAFT [171] proposed to estimate flow fields by 4D correlation volume and recurrent network, yielding an excellent performance. The recent GMFlow [187] turns local correlation matching into global matching, as better results can be achieved with fewer iterations.

Among the various models, relying on pairs of image feature correlations to estimate optical flow is essential. However, as mentioned above, pixel-based correspondence appears to be impaired by motion phenomena such as occlusion, local blurring, etc. In proposal II, we developed a temporal recurrent model that adaptively recognizes current LMC regions, which are jointly refined by introducing spatiotemporal contextual information.

Unsupervised Approaches circumvent the need for labels by optimizing photometric consistency with some regularizations. Yu et al. [193] were the first to present a method for learning optical flow based on the constancy of pixel brightness and motion smoothness. Following several studies improved the performance step-by-step, such as adding occlusion reasoning [110, 182], multi-frame extension [53, 71, 168], epipolar constraints [198], 3D geometrical constraints [191, 200] and stereo depth [86, 181]. Based on self-supervised learning, another branch of work improves performance by learning the flow of ambiguous regions in a knowledge distillation manner [87, 88, 90]. There is a growing tendency to improve performance by integrating multiple strategies [77, 99].

Occlusion remains a fundamental issue during unsupervised learning. Usually, an occlusion mask is employed to remove the photometric losses of occluded regions [182]; thus, such regions are never supervised. Some studies have proposed solutions from a 3D geometric perspective. Additionally, using self-supervised learning, DDflow [88] introduced the concept of using rectangular noise patches to simulate occlusion phenomena. This idea was further refined

by Selfflow [90], which employed a superpixel algorithm to generate occlusions with a more natural shape. This strategy of randomly adding noise to images has been widely adopted in recent studies [91, 99, 159].

However, we hold the viewpoint that occlusion is a long temporal process influenced by interactions among multiple moving objects across multiple frames. Abrupt masking through random noise, as utilized in the aforementioned approaches, does not accurately reflect the smooth motions of objects in natural scenes that adhere to the inertia criterion. Furthermore, images contaminated by regional random noise could be considered outliers—outside the distribution of the original dataset—which might potentially degrade network performance when utilizing the original dataset. Considering this issue, we address the problem by modeling smooth and dynamic motion scenarios. We achieve this by extracting sub-objects from the original data distribution and simulating smooth dynamic occlusions based on the Markov property. By implementing a well-designed mixed loss function, we attain complete supervision of both the flows within occluded regions and the occluder per se.

Temporal Dynamic Models based on RNNs are not commonly used for optical flow estimation. A few studies sought to model multi-frame optical flow estimations using RNNs and LSTMs [51, 119]. However, these approaches rely on conventional supervised training and are applicable to only a few synthetic sequences. Several unsupervised learning approaches employ multi-frame data to estimate optical flow [53, 87, 90]. Compared to those networks, proposal II uses a temporally dynamic model that is able to process real-time causal sequences recurrently. Model inference and training are explicitly decoupled in this study; the prediction of the LMC region is optimized via self-supervised training. In particular, existing self-supervised models employ random noise to erase local regions as occlusion hallucinations [77, 87, 159]. This discontinuous occlusion is acceptable for a two-image-based inference network. However, it can potentially impair the proposed temporal recurrent model, which infer continuous motion patterns across extended temporal environments. We accordingly incorporate a continuous motion occlusion environment via the pseudo-label method, allowing the model to learn a more realistic occlusion rule and using the motion history context to predict the optical flow for LMC regions.

3.3 Temporal Dynamic Modelling & Scene Augmentation

Based on the preliminary of the unsupervised learning background in Section 2.4.3, we introduce a lightweight spatiotemporal dual recurrent network and our basic unsupervised training approach, which learns to estimate optical flow from unlabeled temporal causal sequences.

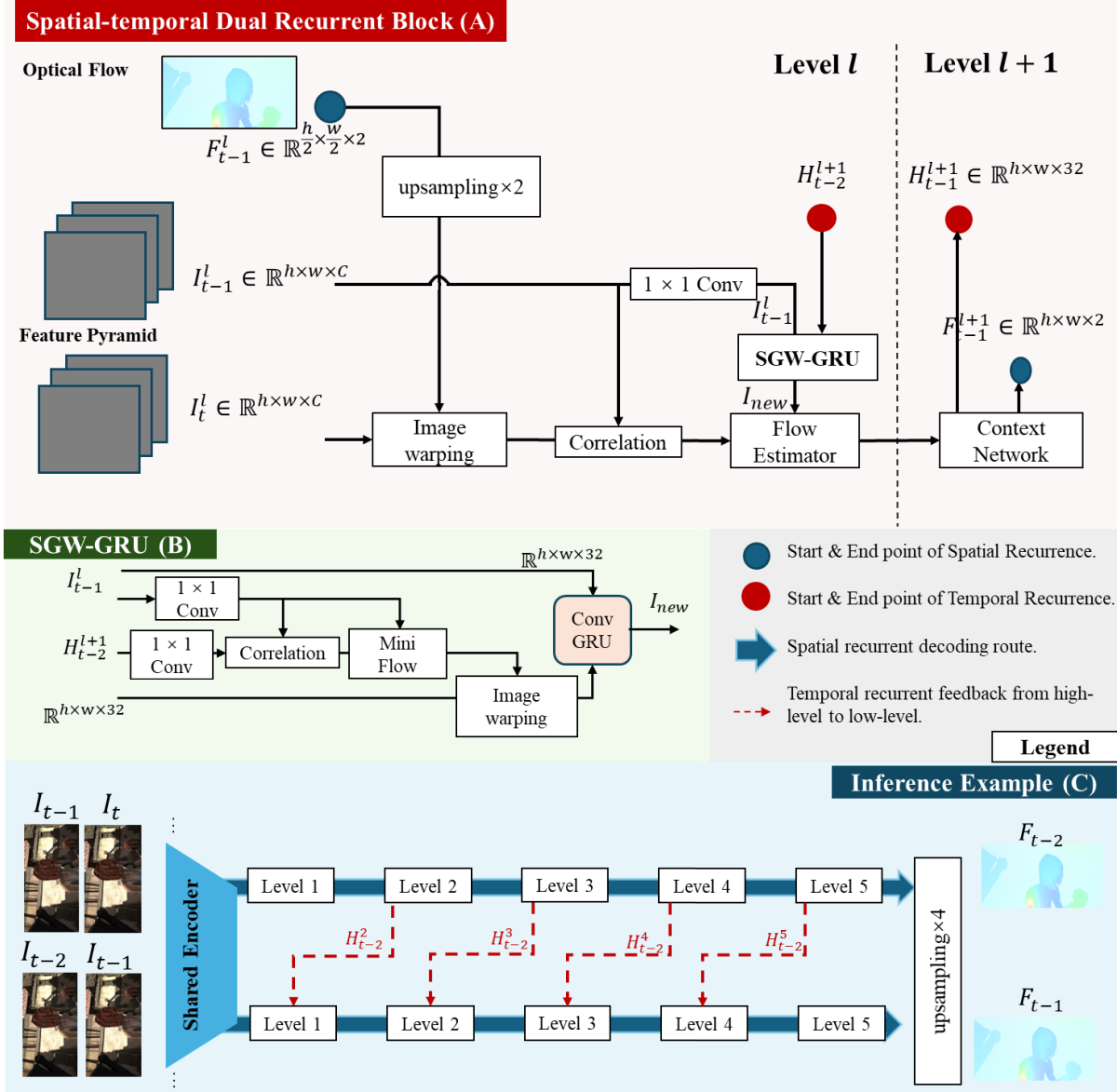


Figure 3.3: Inference architecture of the spatial-temporal dual recurrent block. The spatial recurrence is modified from the basic flow inference pipeline of PWCNet [164], including a cascade of image warping, volume correlation, flow estimation, and context networking. This decodes the flow field from small to large scales. Figure is adapted from [169].

3.3.1 Network structure

We constructed a dynamic RNN that handles arbitrary temporal series $\mathcal{I} = \{I_0, I_1, \dots, I_{t-1}, I_t\}$, receives the previous frame I_{t-1} , current frame I_t , and previous hidden state H_{t-2} as inputs, and outputs the current optical flow F_{t-1} and hidden state H_{t-1} for the next time step. Given a pair of images, a five-level feature map pyramid $\mathcal{P} = \{I^5, \dots, I^1\}$ is generated by a concise feedforward CNN for each image, with the size gradually reducing from $\frac{H}{4} \times \frac{W}{4}$ to $\frac{H}{64} \times \frac{W}{64}$.

To ensure spatiotemporal recurrence, we introduce a convolutional GRU block with self-

guided warping (SGW), i.e., an SGW-GRU block. The SGW-GRU block operates directly on feature pyramids from low- to high-scale in a spatially recurrent manner. Likewise, the block operates across image sequences in a temporally recurrent manner. For a feature pyramid at level l , it has four basic stages of PWCNet [164], i.e., warping $\rightarrow$ correlation volume $\rightarrow$ flow estimator $\rightarrow$ context network, as shown in Fig. 3.3 (A). The difference is that an extra convolutional block is added to the context network to generate a temporal hidden state H_{t-1} characterized by 32-channel feature maps at different scales, which implicitly contain contextual information of motion history during previous moments. The current temporal hidden state is used as input for the next time stage when it is fed back into the SGW-GRU block.

Given moving objects, there are always differences in spatial locations between the features of two adjacent time steps. Accordingly, we use the SGW block to self-adjust the spatial locations of hidden feature maps. In such a block, the correlation between a previously hidden and current feature is first calculated, and a mini-flow estimator is then employed to derive a flow field that guides warping when adjusting the spatial arrangement of H_{t-2}^{l+1} . Finally, a convolutional GRU fuses the hidden state H_{t-2}^{l+1} to the current features I_{t-1}^l and outputs the fused feature I_{new} , as shown in the Fig. 3.3 (B). The module of Conv-GRU applies the basic principle of the original GRU [23] but replaces the core operation with a fully convolutional approach. Specifically, the module accepts the previous high-level hidden state H_{t-2}^{l+1} and current feature I_{t-1} as inputs, and generates a fused feature I_{new} that is input to the flow estimator. This may be formalized as:

$$\begin{aligned}
 z &= \sigma \left(\text{Conv}_{3 \times 3} \left(\text{ConCat} \left[H_{t-2}^{l+1}, I_{t-1} \right] \right) \right) \\
 r &= \sigma \left(\text{Conv}_{3 \times 3} \left(\text{ConCat} \left[H_{t-2}^{l+1}, I_{t-1} \right] \right) \right) \\
 q &= \text{Tanh} \left(\text{Conv}_{3 \times 3} \left(\text{ConCat} [r \odot I_{t-1}, H_{t-2}^{l+1}] \right) \right) \\
 I_{new} &= (1 - z) \odot I_{t-1} + z \odot q.
 \end{aligned} \tag{3.1}$$

where $\sigma(\cdot)$ denotes the sigmoid function; $\text{ConCat}(\cdot)$ the concatenation of feature maps; and $\odot$ the element-wise production.

Regarding temporal recurrence, the previous feedback signal from the deeper layer is transferred to the current timestep to control the coding behaviors. This temporal feedback provides prolonged motion history contextual information, which could facilitate predicting the current motion stage, as demonstrated on the right side of Figure 3.3 (C). Such a design mirrors the temporal predictive coding of the human visual system [95, 142], wherein higher-order neurons send feedback signals to lower-order neurons and thus control their behaviors. Integration of the above modules yields a lightweight network with only 2.50 M parameters, which handles causal sequences of arbitrary length to estimate optical flow.

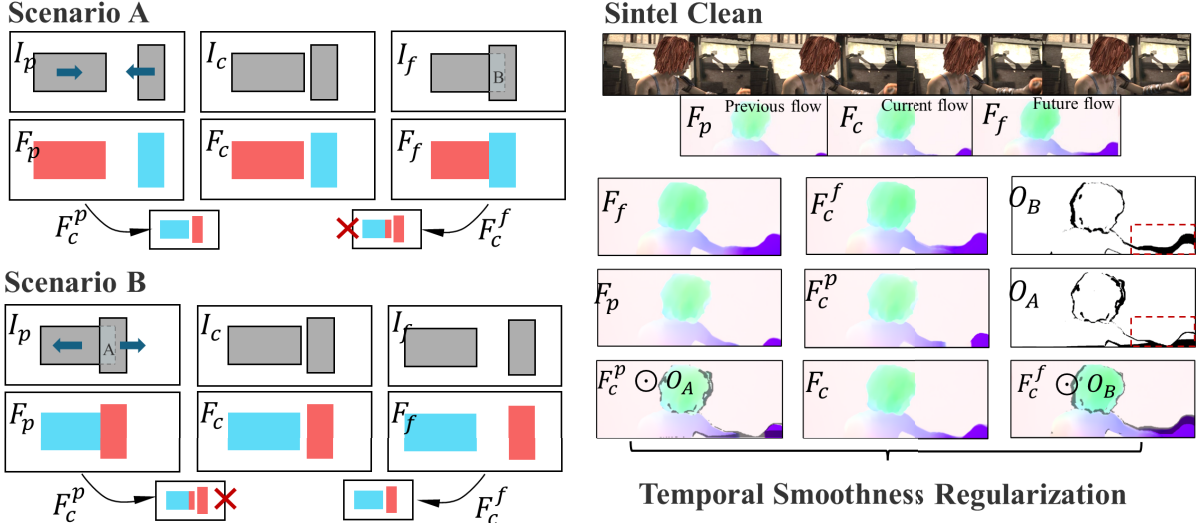


Figure 3.4: Occlusion-aware temporal smoothness regularization. Scenario A: The pixels in a current frame I_c become invisible in the next frame I_f ; Scenario B: The pixels visible in the current frame I_c are invisible in the previous frame I_p . Scenarios A and B cause errors in spatial alignment, which should be considered during temporal smoothness regularization. The right side shows the Sintel Clean scene. Figure is adapted from [169].

3.3.2 Occlusion-aware Temporal Smoothness Regularization.

As mentioned in Section 2.4.3, simply masking the occluded region is inadequate in terms of supervision. Spatial smoothness loss evaluates only low-level RGB similarity. As compensation, we propose temporal smoothness regularization (TSR) to provide more reliable supervision for occluded regions. First, we define four adjacent frames $\{I_{k-1}, I_k, I_{k+1}, I_{k+2}\}$ and three motion flow fields $\{F_p, F_c, F_f\}$, i.e., the forward optical flows of $I_{k-1} \rightarrow I_k$, $I_k \rightarrow I_{k+1}$, and $I_{k+1} \rightarrow I_{k+2}$. Given the limitations imposed by physical laws, the motion trajectory $M(t)$ of an object is always a smooth differential curve, and a video sequence can be regarded as a sampling of the locations thereof. Given a sufficiently short sampling interval Δt , we can create a small temporal window and then derive a Taylor first-order approximation of $M(t)$, i.e., $M(t) = M(t_0) + \Delta t \cdot \frac{dS}{dt}|_{t=t_0}$, or, $M(t) = M(t_0) + t \cdot V(t_0)$, where V can be regarded as a velocity vector or the so-called optical flow within the defined temporal window. With the above assumption, the identical object could be regarded as moving at a similar speed across $\{F_p, F_c, F_f\}$. Thus a triple equation is established based on reverse warping operation:

$$\begin{aligned} \hat{F}_c^f &= F_f(p + F_c(p)), \quad \hat{F}_c^p = F_p(p + F_p^{-1}(p)). \\ \hat{F}_c^p &\approx \hat{F}_c \approx \hat{F}_c^f \end{aligned} \quad (3.2)$$

Where $\hat{F}_c^f$ is the F_c reconstructed from F_f ; $\hat{F}_c^p$ is the F_c reconstructed from F_p . F_p^{-1} denotes the backward flow of $I_k \rightarrow I_{k-1}$. Maintaining the assumption of constant velocity, the above equation yields the basic temporal smoothness constraint. However, importantly, warping is

not reversible and an occlusion problem thus arises during either forward or backward optical flow, as shown on the left of Fig. 3.4. Typically, there are two problematic scenarios. In the first, pixels in the current frame become invisible in the next frame (A). In the second scenario, pixels visible in the current frame are invisible in the previous frame (B). Both scenarios cause errors in spatial alignment. For simplicity, Ω_A, Ω_B indicate two locations where pixels of F_c are invisible in F_p and F_f , respectively. Thus, we generate two occlusion masks O_A, O_B via forward-backward checking:

$$O(\mathbf{p}) = \begin{cases} 1, & \mathbf{p} \notin \Omega \\ 0 & \mathbf{p} \in \Omega \end{cases} \quad (3.3)$$

Temporal smoothness depends on the computation of the differential across time, which, in the discrete case, is equivalent to the computation of the first-order differences across frames. Here, we simplify this by directly using the Charbonnier loss to evaluate the differences between frames, thus ensuring smoothness between F_c and the two adjacent frames. Also, the displacement size of the F_c is utilized as the decay coefficient for temporal smoothing, i.e., locations with more violent motion have smaller smoothing constraints. The final temporal smoothing regularization of the current flow F_c is:

$$\ell_{tsm} \sim \frac{\sum_{\mathbf{p}} \left[\mathbf{C} \left(\mathcal{S}(\hat{F}_c^p), F_c \right) \odot O_A / |F_c| \right]}{\sum_{\mathbf{p}} O_A(\mathbf{p})} + \frac{\sum_{\mathbf{p}} \left[\mathbf{C} \left(\mathcal{S}(\hat{F}_c^f), F_c \right) \odot O_B / |F_c| \right]}{\sum_{\mathbf{p}} O_B(\mathbf{p})}. \quad (3.4)$$

Where $\mathcal{S}(\cdot)$ means to stop the gradient from the computational graph and $\mathbf{C}(\cdot)$ is the Charbonnier loss. At locations $\mathbf{p} \in \{\Omega_A \cap \Omega_B\}$, we do not perform any TSR because the adjacent frames contain no reliable prior flows. At positions $\{\mathbf{p} | \mathbf{p} \in \Omega_A, \mathbf{p} \notin \Omega_B\}$, we add only temporal smoothness from the previous flows F_p . Similarly, temporal smoothness from future flows F_f are available only for positions $\{\mathbf{p} | \mathbf{p} \in \Omega_B, \mathbf{p} \notin \Omega_A\}$; Ultimately, the temporal smoothness values from both F_p and F_f are added to positions $\{\mathbf{p} | \mathbf{p} \notin \Omega_A \cup \Omega_B\}$. The optimized result is that the I_c approximates the linear interpolations between I_p and I_f .

Our design includes a prior assumption. Given the temporal smoothness of motion, Ω_A and Ω_B tend to have only a few intersections, so objects occluded in the next frame are often visible in the previous frame. Therefore, motion information occluded via forward propagation can be supplied by examining past flow. On the lower right side of Fig. 3.4, the background area occluded by the arm in the forward inference is visible in the backward inference; the motion information in the previous frame can be tracked using backward flow as the reference.

3.3.3 Training in the Video Sequence

During each training iteration, we sequentially feed N consecutive frames into the network and simultaneously supervise the $N - 1$ optical flows generated. We employ warping-based image photometric similarity, spatial smoothing loss, and occlusion masking. Note that, as we employ a temporally recursive network, we infer causal sequences and generate optical flow in only one

direction. Therefore, during training, we first infer the forward optical flow of the entire sequence and then reverse the image sequence to infer backward optical flow. Finally, a forward-backward check is performed to obtain the occlusion mask sequence $\{O_t\}_{t=1}^{N-1}$. Specifically, at the end of the sequence, we update the weights to decrease the following loss:

$$\mathcal{L} = \frac{1}{N-1} \sum_{t=1}^{N-1} \mathcal{L}_t \quad (3.5)$$

where $\mathcal{L}_t$ is the combination of the multi-scale loss for image pair (I_t, I_{t+1}) and the temporal smoothness loss:

$$\mathcal{L}_t = \lambda_1 \ell_{sm}(F_t) + \lambda_2 \ell_{tsm}(F_{t-1}, F_t, F_{t+1}) + \ell_{warp}(I_t, I_{t+1}, F_t, O_t). \quad (3.6)$$

The λ_1 and λ_2 hyper-parameters balance the loss weights of the loss function.

3.3.4 Dynamic Scene Augmentation

As indicated by DDFlow [89], self-supervised learning-based distillation effectively improves unsupervised optical flow estimations and has been widely adopted [77, 87, 90, 99]. Generally, a teacher model $f_t(\cdot)$ processes original pairs of samples $\mathcal{I} = \{I_a, I_b\}$ to generate the optical flow as an pseudo-label F_a^* that trains a student model $f_s(\cdot)$. Before $\mathcal{I}$ is fed to $f_s(\cdot)$, a series of specific image transformations $\mathcal{T}_I(\cdot) : \mathcal{I} \mapsto \tilde{\mathcal{I}}$ are used to increase scenic diversity. Such data augmentation generates a lower-confidence optical flow F_a , which is then self-supervised using the pseudo-label from $f_t(\cdot)$. To ensure spatial consistency, the pseudo-label must be consistently transformed using $\mathcal{T}_F(\cdot) : F^* \mapsto \tilde{F}^*$. Then, self-supervision based on the pseudo-label is achieved by considering the Charbonnier similarity between pseudo-label $\mathcal{T}_F(F_a^*)$ and F_a , thus:

$$\ell_{self} \sim \mathbf{E}_{\mathbf{p}} \left(\left| \mathcal{S}(F_a(\mathbf{p})) - \tilde{F}_a^*(\mathbf{p}) \right| + \epsilon \right)^q \quad (3.7)$$

Here, we extend this strategy to multi-frame sequences and simulate multiple dynamic natural variations. The network comes to understand occlusion and variation in a dynamic environment. Specifically, we use three training enhancers, i.e., a DOE, an SVE, and a CVE, as described below.

Dynamic occlusion enhancer

We aim to simulate more naturalistic occlusion hallucinations in the DOE. There are two steps, the first of which is (i) random cropping. This operation is critical as, in videos, objects often move out of the frame due to their inherent motion, resulting in an inability to find correspondence between consecutive frames, which could be termed "out-of-boundary" occlusions. By reducing images to a smaller dimension through cropping, we can simulate this type of occlusion. We crop the image sequences during preprocessing prior to occlusion transformation. The second step is (ii) dynamic occlusion simulation. Given a set of frames, we divide the images

into multiple subregions using a super-pixel segmentation algorithm [162]. Subsequently, n subregions are randomly selected as “occluders”. Rather than adding random noise, we create a natural texture for each occluder by extracting the textures of the original sample batches. Finally, we simulate occlusions by randomly placing n occluders into the first frame of a sequence. Unlike previous work, we simulate the motion of the occluders dynamically in multi-frame se-

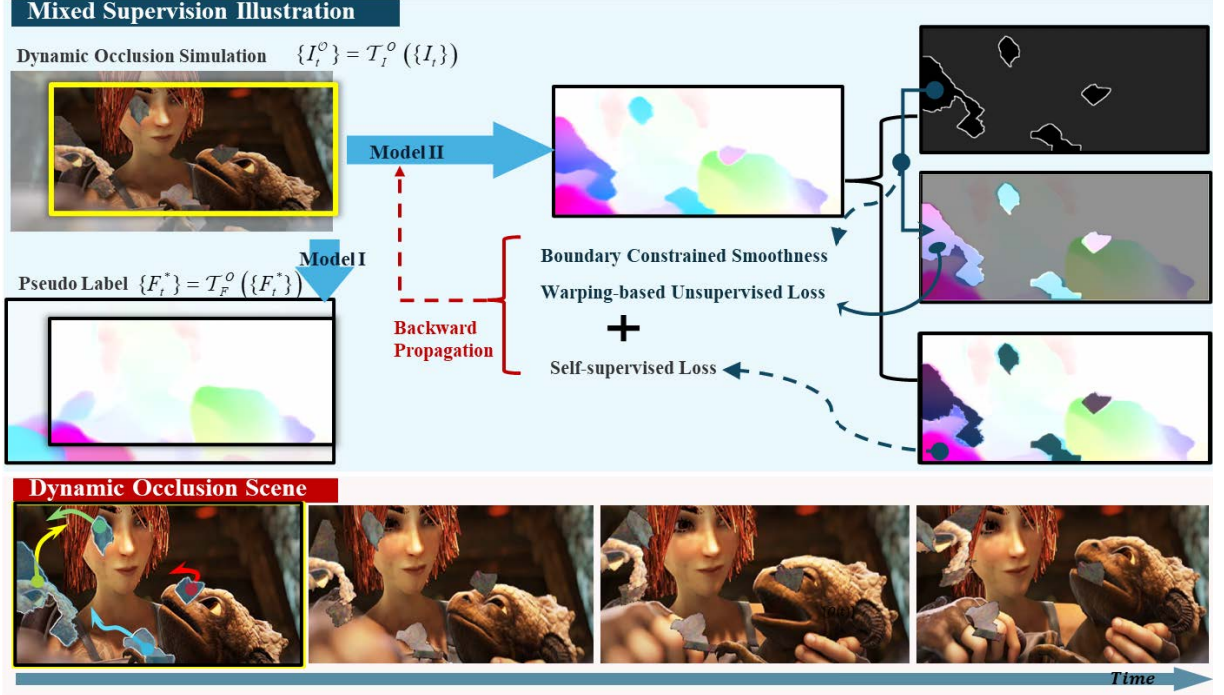


Figure 3.5: Mixed supervision strategy with dynamic occlusion. To simulate natural object motion, we use a super-pixel algorithm to extract subobjects of images that serve as naturally textured artificial “occluders”, and then apply a Markov process to simulate the smoothness of random motion, the effect of which is shown in the bottom of the figure (where the occluders are in yellow). Using the known prior occlusion masks, we combine self-supervised and unsupervised learning to ensure mixed supervision (blue box). Figure is adapted from [169].

quences. Obviously, objects’ motion always keeps smoothness in natural environments, and the object’s motion at the next time step depends heavily on the motion state of the current moment. To better simulate random motions while keeping the above rule, we partially follow the Markov chain principle to simulate the process. Specifically, we assume the moving state of an object as a random process $\mathbf{S}(t)$, which can be decomposed into two orthogonal vectors $\mathbf{S}(t) = [U(t), V(t)]$. For any time step t , we introduce Markov properties by assuming its moving state only depends on the motion state of the previous time step $t - 1$, i.e.:

$$\begin{aligned} \Pr[\mathbf{S}(t) = s_t \mid \mathbf{S}(t-1) = s_{t-1}, \mathbf{S}(t-2) = s_{t-2}, \dots] \\ = \Pr[\mathbf{S}(t) = s_t \mid \mathbf{S}(t-1) = s_{t-1}] \end{aligned}$$

The motion states $[U, V]$ at time t are sampled from a two dimensional Gaussian distributions: $[U(t), V(t)] \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where we let

$$\boldsymbol{\mu} = \begin{pmatrix} \mu_U(t) \\ \mu_V(t) \end{pmatrix}, \quad \boldsymbol{\Sigma} = \begin{pmatrix} \sigma_U^2 & 0 \\ 0 & \sigma_V^2 \end{pmatrix},$$

by simplistically assume an independent relation between $U(t)$ and $V(t)$. σ_U, σ_V are set as a constant value that controls the variation of the moving process, and the mean μ_U, μ_V are set as equal to the motion state at the previous moment, i.e., $[\mu_U(t), \mu_V(t)]^T = [U(t-1), V(t-1)]^T$, by which a random motion process while keeping smoothness is simulated for each artificial occluder.

The initial moving state $\mathbf{S}(0)$ is also defined as a random variable, where moving speed $|\mathbf{S}(0)|$ is sampled from a 1-D Gaussian distributions $\mathcal{N}(\mu, \sigma)$, and moving angle is sampled from $\mathbf{U}(0, 2\pi)$. To fit the distribution of the dataset, μ is set to be equal to the statistical expectations of the moving speed across the whole training dataset and $\sigma = \frac{\mu}{3}$.

In summary, given a sequence of images $\{I_t\}$, the process of dynamically simulating occlusion consists of the following steps: 1). Randomly cropping the image sequence. 2). Randomly selecting a keyframe I_k from $\{I_t\}$. 3). Segmenting I_k into sub-regions by the super-pixel segmentation algorithm. 4). Randomly extracting N sub-regions $\{\Psi_n\}_{n=1}^N$ and creating new textures. 5). Randomly assigning $\{\Psi_n\}$ at any location in the first frame I_0 and initialing the moving state. 6). Starting from the I_0 , the 0.85damic masking process is simulated based on Markov's principle.

In the actual motion implementation, we simulate not only the object's translation but also the object's rotation, scaling, and distortion, which all obey the proposed Markov process to ensure relatively smooth motion characteristics.

Mixed supervision

At the moment t , each occluder $\{\Psi_n^t\}_{n=1}^N$ is regarded as the closest object to the lens at the highest level of occlusion. It is then not difficult to generate the occlusion mask O_t :

$$O_t(\mathbf{p}) = \begin{cases} 1, & \mathbf{p} \notin \Omega_\Psi \\ 0, & \mathbf{p} \in \Omega_\Psi \end{cases}$$

where $\Omega_\Psi = \{\Psi_1^t \cup \Psi_2^t \cup \dots \cup \Psi_N^t\}$. As shown in Fig. 3.5, we designed a mixed supervision strategy to drive model learning occlusion laws in simulated dynamic occlusion sceneries, formalized as:

$$\ell_{doe}(t) = \ell_1(t) + \ell_2(t), \quad (3.8)$$

where ℓ_1 avoids losses caused by the occluders, using the pseudo-label to supervise regions without occluders:

$$\ell_1(t) \sim \mathbf{E}_{\mathbf{p}} \left(\mathbf{C} \left(\widetilde{F}_t^*(\mathbf{p}), F_t(\mathbf{p}) \right) \odot O_t(\mathbf{p}) \right), \quad \mathbf{p} \notin \Omega_\Psi. \quad (3.9)$$

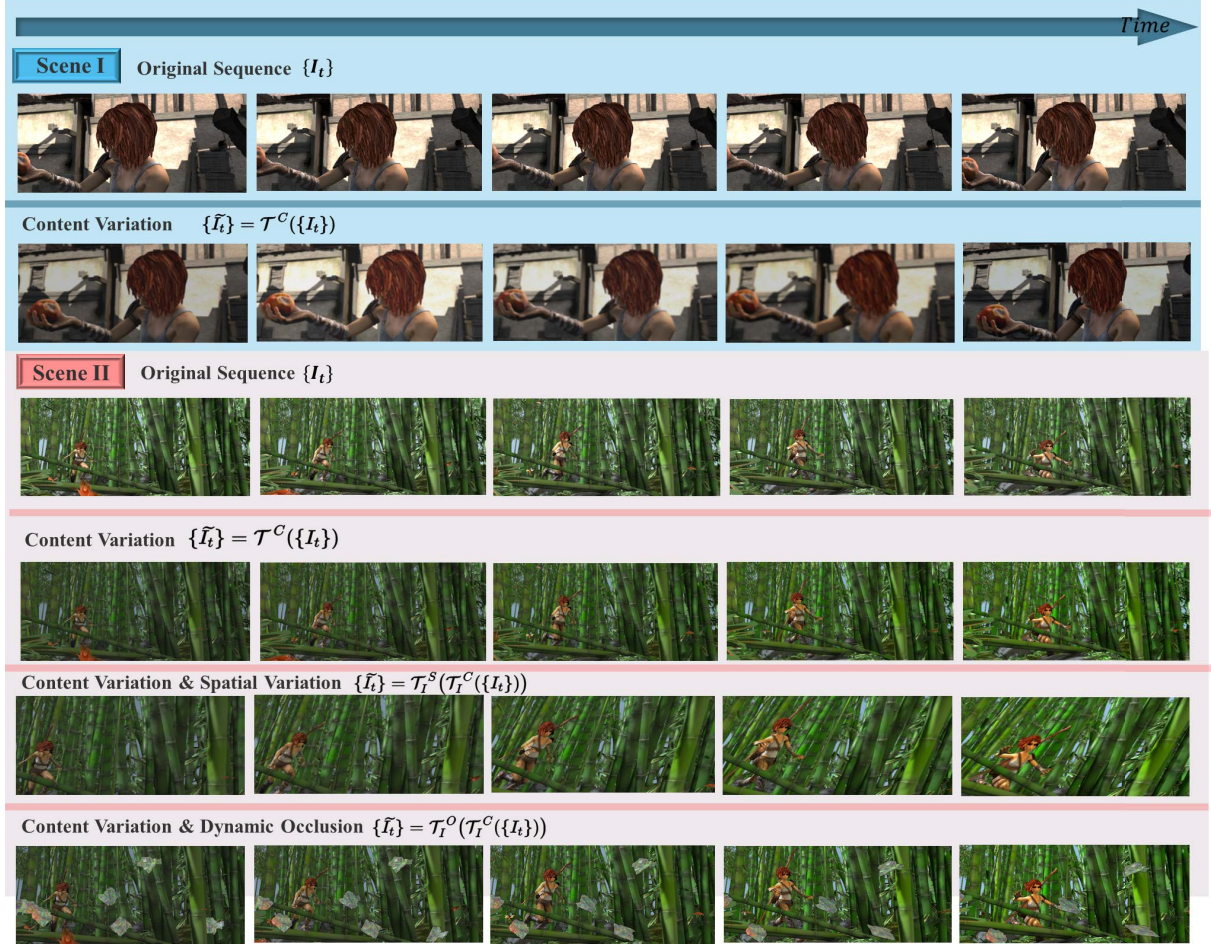


Figure 3.6: Qualitative demonstration of temporal dynamic variations, including content, spatial variations, and dynamic occlusions. We perform dynamic transformations on the original image sequence (A), such as the monotonic increase in chromaticity and illumination in (B); dynamic shifts and warps in spatial positions (C); and dynamic movements of the artificial occluders in a Markov process (D). Using self-supervised distillation, all of these dynamics are well-handled by the model. Figure is adapted from [169].

ℓ_2 is an unsupervised loss based on image warping, which is available for the regions $\mathbf{p} \in \Omega_\Psi$:

$$\ell_2(t) \sim \text{SSIM}(\mathcal{W}_t(I_{t+1}), I_t) \odot (1 - O_t) + \ell_{sm}, \quad (3.10)$$

where $\text{SSIM}(\cdot)$ operates only on regions with occluders by the mask $1 - O_t$. Here, ℓ_{sm} is similar to equation 2.7, but the image gradient-based attenuation term is modified by replacing the image I_t with the occlusion mask O_t , thus restricting smoothing regularization within sub-regions of each occluder. The DOE with mixed loss strategy are evaluated in the ablation studies (see Table 3.5).

Spatial & Content Variation Enhancements

The SVE implements a series of spatial transformations, including random sequence cropping, rotation, horizontal and vertical flipping, and thin-plate-spline or CPAB transformations [46], which can actually be regarded as spatial data augmentation. We extend the spatial transformations to continuous time series and simulate them across time. Natural image sequences exhibit spatial jitter and distortion attributable to environmental vibrations and camera shake. Several such challenging scenes are included in the Sintel Final dataset [18]; we generalized the spatial variations by applying a series of random affine transformations along the sequence to simulate variations and performed object rotation, distortion, scaling, and translation. Such spatial transformations (except cropping) shift the pixel positions of the entire sequences by $\mathcal{T}_I^S : \{I_t\} \mapsto \{\tilde{I}_t^S\}$, which can also be equalized as:

$$\{\tilde{I}_t^S(\mathbf{p})\} = \{I_t(\tau_{\theta_t}(\mathbf{p}))\}, \quad (3.11)$$

where $\tau_{\theta_t}(\cdot)$ is the transformation of the pixel coordinates of image I_t using the transform parameters of θ_t . To maintain consistency between the transformed scenes and pseudo-label F_t^* , the optical flows must undergo different transformations because the application of different transformation parameters to I_t and I_{t+1} varies the optical flow fields. Here, we first track the flow variation between $\{\tau_{\theta_t}, \tau_{\theta_{t+1}}\}$ via inverse-affine transformation, superimpose this on the offset of the original flow field F_t^* , and finally use $\tau_{\theta_t}(\cdot)$ to ensure spatial consistency with $\tilde{I}_t$. The entire process $\mathcal{T}_F^S : \{F_t^*\} \mapsto \{\tilde{F}_t^*\}$ can be formalized as:

$$\begin{cases} F_{\text{new}}(\mathbf{p}) = \tau_{\theta_{t+1}}^{-1}(\mathbf{p} + F_t^*(\mathbf{p})) - \tau_{\theta_t}^{-1}(\mathbf{p}) \\ \tilde{F}_t^*(\mathbf{p}) = F_{\text{new}}(\tau_{\theta_t}(\mathbf{p})) \end{cases} \quad (3.12)$$

Given the pseudo-label $\tilde{F}_t^*(\mathbf{p})$, the loss function based on self-supervised learning by the SVE is:

$$\ell_{sve}(t) \sim \mathbf{E}_{\mathbf{p}} \left(\mathbf{C} \left(\tilde{F}_t^*(\mathbf{p}), F_t^S(\mathbf{p}) \right) \right) \quad (3.13)$$

Similar to the SVE, the CVE performs various transformations of image content. First, we introduce color jitter following basic data augmentation principles [87], followed by random brightness, saturation, hue, and gamma transformation. Gaussian blurring and noise are randomly introduced over the entire image sequence to render the scenes more challenging.

Aside from static augmentation, the contents of natural scenes vary greatly, including changes in brightness and hue as illumination varies. We simulated such scenarios by monotonically increasing or decreasing illumination, saturation, hue, and random jitter over time. Besides, we simulated regional linear motion blur and camera defocus blur across frames, thus forcing the network to focus on structural object information rather than pixel differences.

Specifically, the content transformation of a sequence of images can be written as $\mathcal{T}_I^C(\cdot) : \{I_t\} \mapsto \{\tilde{I}_t^C\}$. In a relatively simple case, $\mathcal{T}_I^C(\cdot)$ neither changes the locations of pixels nor introduces a new occlusion. Therefore, no extra transform is required for the corresponding

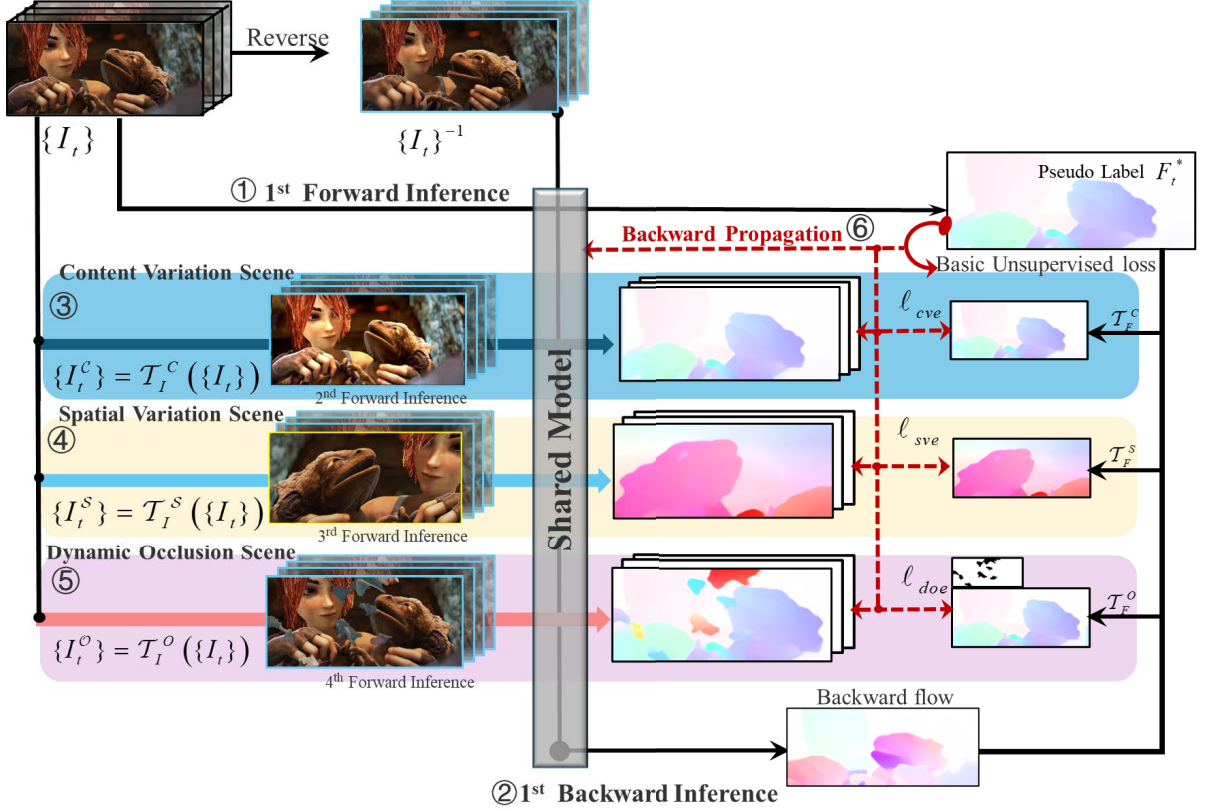


Figure 3.7: Complete training pipeline based on self-supervised learning. Three dynamic training enhancers expand the original sequence to scenes wherein the content, spatial position, and dynamic occlusion vary. Four forward inferences and one backward inference are performed during each iteration: the first forward and backward inferences are used for basic unsupervised training on original scenes and computation of confidence masks. The final three forward inferences evaluate extended scenes and are back-propagated using a self-supervised loss function. The entire process relies on long dynamic sequences, allowing the network to understand variations and occlusions in a dynamic environment. The execution order is marked from ① to ⑥. Figure is adapted from [169].

pseudo-label, and the self-supervised loss of the content variation training enhancer can be expressed as:

$$\ell_{cve}(t) \sim \mathbf{E}_{\mathbf{p}} \left(\mathbf{C}(F_t^*(\mathbf{p}), F_t^C(\mathbf{p})) \right) \quad (3.14)$$

Qualitative illustrations of the three transformations $\mathcal{T}_I^O(\cdot)$, $\mathcal{T}_I^S(\cdot)$, and $\mathcal{T}_I^C(\cdot)$ are shown in Fig. 3.6.

3.3.5 Self-supervised Distillation

During self-supervised learning, a teacher model generates pseudo-labels to supervise the predictions of a student model. However, the process can be simplified to online learning based on a single model that acts as both the teacher and student. We adopt the strategy of ARFlow [87]

by running an extra forward pass on each transformed scene using the transformed pseudo-label as the self-supervision signal. We first expand all scenes using the previously defined transformations $\{\mathcal{T}_I^O(\cdot), \mathcal{T}_I^S(\cdot), \mathcal{T}_I^C(\cdot)\}$ to obtain the three scenes $\{\widetilde{I}_t^O\}, \{\widetilde{I}_t^S\}, \{\widetilde{I}_t^C\}$ based on the original scene sequence $\{I_t\}$. Training involves four forward inferences, the first of which generates the optical flow pseudo-label $\{F_t^*\}$ of the original scene; the subsequent three yields the optical flow of the dynamic occlusion, spatial variation, and content variation respectively, which then undergo flow transformation $\{\mathcal{T}_F^O(\cdot), \mathcal{T}_F^S(\cdot), \mathcal{T}_F^C(\cdot)\}$ to ensure appropriate image transformation. By combining the loss functions of Eq. 3.6, Eq. 3.8, Eq. 3.13, and Eq. 3.14, the final loss function for self-supervised distillation can be formalized as:

$$\mathcal{L}_{self} \sim \frac{1}{N-1} \sum_{t=1}^{N-1} \underbrace{\mathcal{L}_t}_{\text{1st Infer}} + \Lambda \cdot \underbrace{[\ell_{doe}(t), \ell_{sve}(t), \ell_{cve}(t)]}_{\text{2nd, 3rd, 4th Infer}}. \quad (3.15)$$

where $\Lambda = [\lambda_3, \lambda_4, \lambda_5]^T$ are the combined weight parameters for self-supervised loss. By back-propagating $\mathcal{L}_{self}$, we achieve both basic unsupervised optical flow learning and self-supervised distillation. The overall pipeline of the self-supervised distillation training is illustrated in Fig 3.7. Another alternative approach involves applying all three enhancement techniques simultaneously to the video sequence. However, we chose to treat these scenarios separately due to concerns about potential complexity and loss of control. The combination of all these techniques could lead to a situation that is overly intricate and difficult to manage. For example, CVE might distort the color appearance of the simulated moving occluder, introducing errors into the unsupervised loss function. Furthermore, while DOE involves cropping the image to a smaller size to simulate boundary occlusion, both CVE and SVE retain the original image dimensions. Handling these techniques separately facilitates augmenting multi-scale samples in training.

3.4 Experiments I

We performed comprehensive experiments using several well-established datasets including MPI-Sintel [18], KITTI 2012 [47], KITTI 2015 [114], and FlyingChair [38]. MPI-Sintel and KITTI were used to validate the performance of our model compared to state-of-the-art methods. The FlyingChair dataset was used for model pre-training.

3.4.1 Implementation Details

Using Eq. 3.6 and Eq. 3.15, we set the loss function weights $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5\}$ to $\{50, 0.005, 0.3, 0.3, 0.3\}$ for the Sintel dataset and $\{75, 0.001, 0.2, 0.2, 0.2\}$ for the KITTI dataset, respectively, which follows previous optical flow estimation studies [87]. We implemented our method using PyTorch on a workstation with four parallel RTX A6000 GPUs running CUDA 10.1. All models were trained by the Adam optimizer [80] with $\beta_1 = 0.9$, $\beta_2 = 0.99$. We pre-trained the models on

Table 3.1: Comparison on the Sintel Benchmark. We employed the average EPE error (“the lower the better”) as the evaluation metric for the Sintel datasets. Missing entries ‘-’ indicate that the results are not reported in the publication, and $(\cdot)$ indicates test images used during unsupervised training. ‘*’ highlights the semi-supervised methods.

	Method	Sintel Train		Sintel Test		Param.
		Clean	Final	Clean	Final	
supervised	FlowNetS [38]	4.50	5.45	7.42	8.43	32.07M
	LiteFlowNet [65]	(1.64)	(2.23)	4.86	6.09	5.37M
	PWCNet [164]	(2.02)	(2.08)	4.39	5.04	8.75M
	IRR-PWC [67]	(1.92)	(2.51)	3.84	4.58	6.36M
	SelFlow-ft [90]	(1.68)	(1.77)	3.74	4.26	4.79M
	STARFlow [51]	(2.10)	(3.49)	2.72	3.71	4.77M
	RAFT [171]	(0.77)	(1.20)	2.08	3.41	5.26M
un/semi-supervised	UnFlow [110]	-	(7.91)	9.38	10.22	116.58M
	OAFlow [182]	(4.03)	(5.95)	7.95	9.15	5.12M
	MFOccFlow [71]	(3.89)	(5.52)	7.23	8.81	12.21M
	DDFlow [89]	(2.92)	(3.98)	6.18	7.40	4.27M
	SelFlow [90]	(2.88)	(3.87)	6.56	6.57	4.79 M
	ARFlow [87]	(2.79)	(3.73)	4.78	5.89	2.24 M
	SimFlow [69]	(2.86)	(3.57)	5.92	6.92	9.70M
	UFlow [77]	(2.50)	(3.39)	5.21	6.50	-
	UPFlow [99]	(2.33)	(2.67)	5.24	6.50	3.49M
	OIFlow [91]	(2.44)	(3.35)	4.26	5.71	5.26M
	ALFlow* [195]	(1.94)	(2.55)	3.96	5.35	2.24M
	ULDENet (Ours)	(2.20)	(2.65)	3.50	4.70	2.50M

FlyingChair with a batch size of eight and learning rate of $1e^{-4}$, followed by formal training on the Sintel or KITTI dataset. In the first stage, we used the basic unsupervised loss function with a sequence length of six, batch size of four, and learning rate of $1e^{-4}$ over 100 epochs of training. In the second stage, we performed 50 epochs of intensive training using self-supervised distillation, where the sequence length was eight per iteration, the batch size was four, and the learning rate was $5e^{-5}$ for Sintel and $8e^{-5}$ for KITTI. Gradient clipping was applied during all training to accelerate convergence.

In terms of preprocessing, images from the Sintel dataset were cropped to 384×832 and

Table 3.2: Comparison on KITTI Benchmark. KITTI 2015 Test uses the Percentage of Optical Flow Outliers (FI) as the evaluation metric. Others are employed with the average EPE as the evaluation metric (the lower, the better). ‘-’ indicates that the results were not reported in the publication, and (·) indicates test images used during unsupervised training. ‘*’ highlights the semi-supervised methods.

	Method	KITTI 2012		KITTI 2015		Param.
		Training	Test	Training	Test	
supervised	FlowNet2-ft [68]	1.28	1.8	2.30	11.48%	162.5M
	PWCNet [164]	1.45	1.7	2.16	9.60%	8.75M
	IRR-PWC [67]	-	-	1.63	7.65%	6.36M
	SelFlow-ft [90]	0.76	1.5	1.18	8.42%	4.79M
	STARFlow [51]	-	-	-	7.65%	4.77M
	RAFT [171]	-	-	0.64	5.27%	5.26M
un/semi-supervised	UnFlow [110]	3.29	-	8.10	23.3%	116.58M
	OAFlow [71]	3.55	4.2	8.88	31.2%	5.12M
	DDFlow [89]	2.35	3.0	5.72	14.29%	4.27M
	SelFlow [90]	1.69	2.2	4.84	14.19%	4.79M
	EpiFlow [198]	(2.51)	3.4	(5.55)	16.95%	8.75M
	NLFlow [173]	3.02	4.5	6.05	22.75%	-
	ARFlow [87]	1.44	1.8	2.85	11.8%	2.24M
	UFlow [77]	1.68	1.9	2.71	11.13%	-
	UPFlow [99]	1.27	1.4	2.45	9.38%	3.49M
	OIFlow [91]	1.33	1.6	2.57	9.81%	5.26M
	ALFlow* [195]	(1.22)	1.5	(2.29)	9.65%	2.24M
	Ours	(1.15)	1.3	(2.23)	9.13%	2.50M

inferred to be at the original sizes during validation; the images in the KITTI dataset were resized to 256×832 for both training and validation. We used random image horizontal flipping and sequence reversion to augment the datasets. During training, we mixed Clean, Final, and Albedo scenes from the Sintel training set. We employed the multi-view extension version of the KITTI dataset for sequence training and the flow ground truths were used only for validation. The standard average endpoint error (EPE) and percentage of erroneous pixels (F1) served as evaluation metrics of optical flow. During the validation of the Sintel dataset, we dynamically input all frames of each scene into the model and validated the results for each frame. During

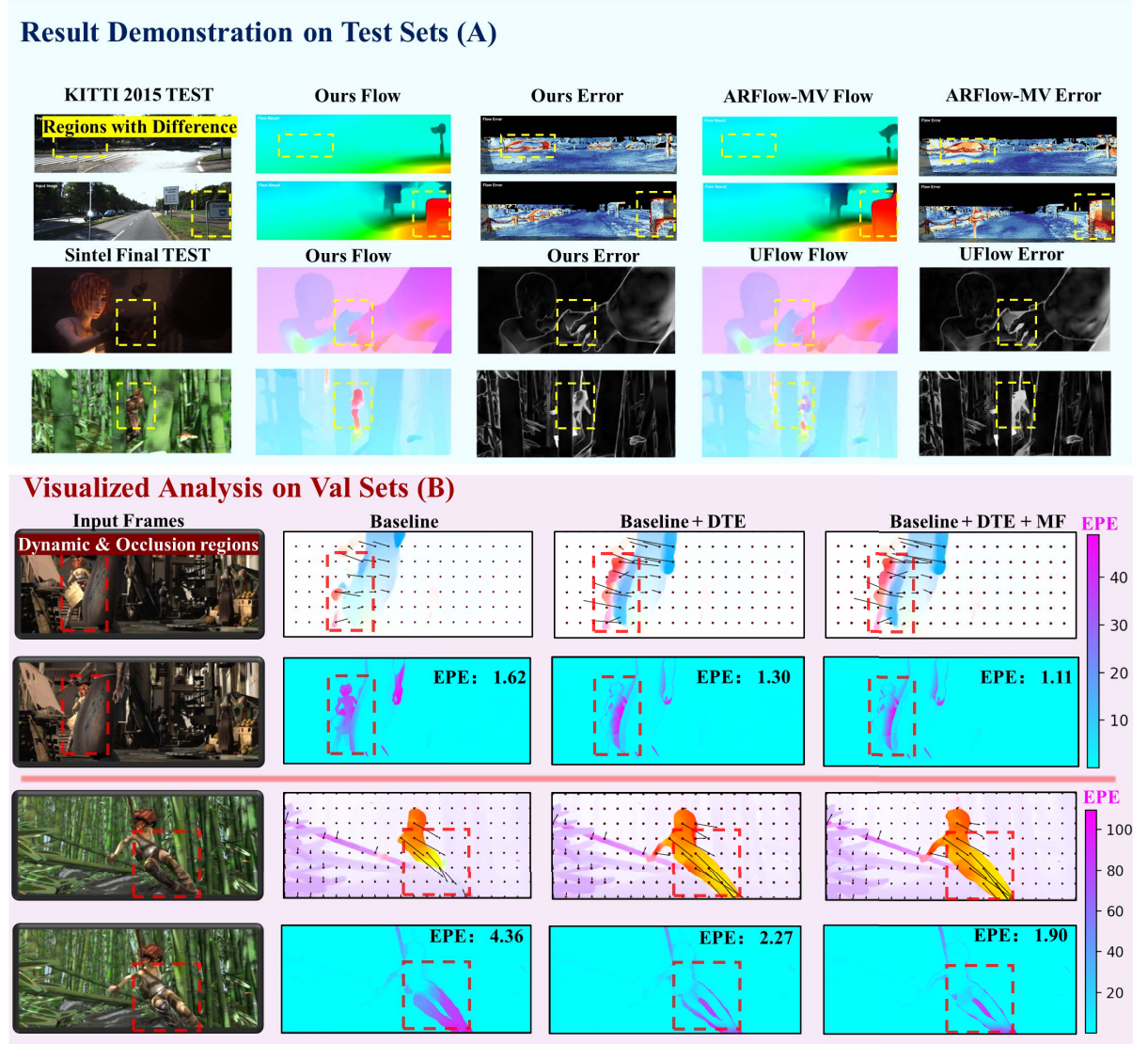


Figure 3.8: (A) Visualization of results on Sintel and KITTI test benchmark. We qualitatively compared the results to those of ARFlow-MV [87] and UFlow [77], where the red box indicates the region wherein our method dominates. Quantitative comparisons are shown in Tables 3.1 and 3.2. All results were provided by the official test server; (B) Visualized analysis on Sintel val set. Baseline: ULDENet with basic unsupervised learning. DTE: dynamic training enhancer. MF: multi-frame inference. The figure demonstrates that DTE significantly improves performance in dynamic and occluded regions. Figure is adapted from [169].

the validation of the KITTI dataset, we input frames 1st–11th and validated the results of frame 10th only.

3.4.2 Comparison of the Sintel & KITTI benchmarks

We compared our method with both supervised and unsupervised approaches using the Sintel Clean and Final datasets. The test results were validated by the official Sintel website. In Table

3.1, our method achieves the highest accuracy on test datasets while maintaining relatively low parameter overheads. Specifically, on the Sintel test set, our results for Clean scenes surpassed those of the previous OIFlow [91], registering an 18% decrease in the EPE metric with only a 48% memory overhead. When juxtaposed with the latest semi-supervised method, ALFlow [195], our model exhibited improved generalization performance with a comparable number of model parameters and inference speed (ours: 2.50M, 17ms/frame; ALFlow: 2.24M, 16ms/frame).

Furthermore, our method was evaluated using both the KITTI 2012 and KITTI 2015 benchmarks. The quantitative results are presented in Table 3.2. When pitted against other unsupervised learning methods, ULDENet displayed superior EPE performance. On the KITTI 2015 dataset, our approach surpassed UPFlow [99], achieving a 9% error reduction in EPE (from 2.45 to 2.23) while using only 71% of the UPFlow parameters. For the KITTI 2015, we reduced the F1-all value from UFlow’s [77] 11.1% to 9.1%, an 18.0% error reduction. On the KITTI 2012, ULDENet, when applied to the training set, reduced UPFlow’s 1.27 EPE to 1.15, amounting to a 9% error reduction.

Qualitative comparisons for the Sintel and KITTI test benchmarks, corresponding to the table values, are displayed in Fig. 3.8 (A). Our model’s gains are marked in the yellow dashed box, which typically highlights regions prone to occlusions and dynamic variations. Fig. 3.8 (B) delves deeper into this phenomenon by individually visualizing the results of different model configurations. The baseline exhibits inadequate results in dynamic and occlusion zones, typical challenges for optical flow estimation. Our proposed Dynamic Training Enhancer (DTE) significantly alleviates such errors. This is because DTE simulates multiple dynamic variation scenarios and occlusion phenomena in self-supervised training, offering the model the correct responses to navigate these challenging scenes. Furthermore, multi-frame inference refines flow predictions by incorporating additional temporal context. The final performance of our proposed model, as shown in Tables 1 and 2, results from the synergy of the various components we introduced. In the subsequent ablation study section, we offer a more detailed analysis and discussion.

3.4.3 Ablation Study

An extensive ablation study was conducted. The Sintel training set was subdivided into separate training and validation sets using [87]. AEPE errors for all pixels (all), non-occluded pixels (NOC) and occluded pixels (OCC) are reported.

Main ablation

All components that we used, including the multi-frame recurrent inference structure (MFI), TSR, and the dynamic training enhancer (DTE), were comprehensively verified using the Sintel dataset. As indicated in Table 3.3, ‘Base’ represents the baseline, i.e., the original lightweight PWCNet operating in the double-frame inference mode. The MF achieves multi-frame training

Table 3.3: Main ablation study on different combinations of the proposed components. Base: baseline of the original lightweight PWCNet with double-frame inference. MF: multi-frame training and inference structure. TSR: temporal smoothness regularization. DTE: dynamic training enhancer.

Base	MF	TSR	DTE	Sintel Clean			Sintel Final			Param.
				ALL	NOC	OCC	ALL	NOC	OCC	
✓				2.23	0.97	19.71	3.23	1.80	23.69	2.10M
✓	✓			2.17	0.91	19.36	3.12	1.74	22.86	2.50M
✓	✓	✓		2.10	0.88	18.74	3.06	1.70	22.51	2.50M
✓	✓		✓	1.75	0.81	15.21	2.81	1.63	19.83	2.50M
✓	✓	✓	✓	1.67	0.77	14.54	2.76	1.61	19.37	2.50M

and inference via the spatial-temporal dual recurrent block, which greatly improves performance but increases the number of parameters by 0.4 M. After adding TSR, the model becomes more robust in occluded regions. TSR more reliably deals with occluded regions by exploring prior motion in adjacent frames, and the EPE of OCC regions in the clean dataset falls from 19.36 to 18.74. DTE significantly improves network performance without the need for additional memory or computational overhead. Quantitatively, the network performance improves from an EPE of 2.17 to 1.75, with a 21% decrease in the EPE of OCC regions. After integrating all strategies, we eventually improved performance from an EPE of 2.23 to 1.67 for the Clean dataset, and from 3.23 to 2.76 for the Final dataset, thus reducing the EPE by an average of 20%.

Network Structure

Table 3.4 concisely compares models with different structures. To make a fair comparison, we used the same loss strategy of ARFlow for all models. Our network substantially outperformed PWCNet [164] and required fewer network parameters. Compared to the multi-frame approach ARFlow-MV [87], which extracts five adjacent frames before and after the current time, we only input frames before the current time, thus allowing ULDENet to be fitted to a real-time task. Also, our method handles sequences of arbitrary lengths, unlike most static multi-frame methods such as ARFlow-MV, SelFlow [90] and MFOccFlow [71]. Quantitatively, we significantly reduced the EPE at the cost of a slight increase in memory overhead.

Although our model is trained in a dynamic environment with multiple frames, it still shows good generalization in terms of two-frame inference after convergence. Specifically, we split each scene of the Sintel dataset into different frames to verify the accuracy w.r.t. the inference frames; the results are shown in Fig. 3.9. The EPE of the model gradually decreases as the number of inference frames increases, indicating that multi-frame inference increases the gain on the Final dataset with more complicated scenes, in turn suggesting that ULDENet is optimal when

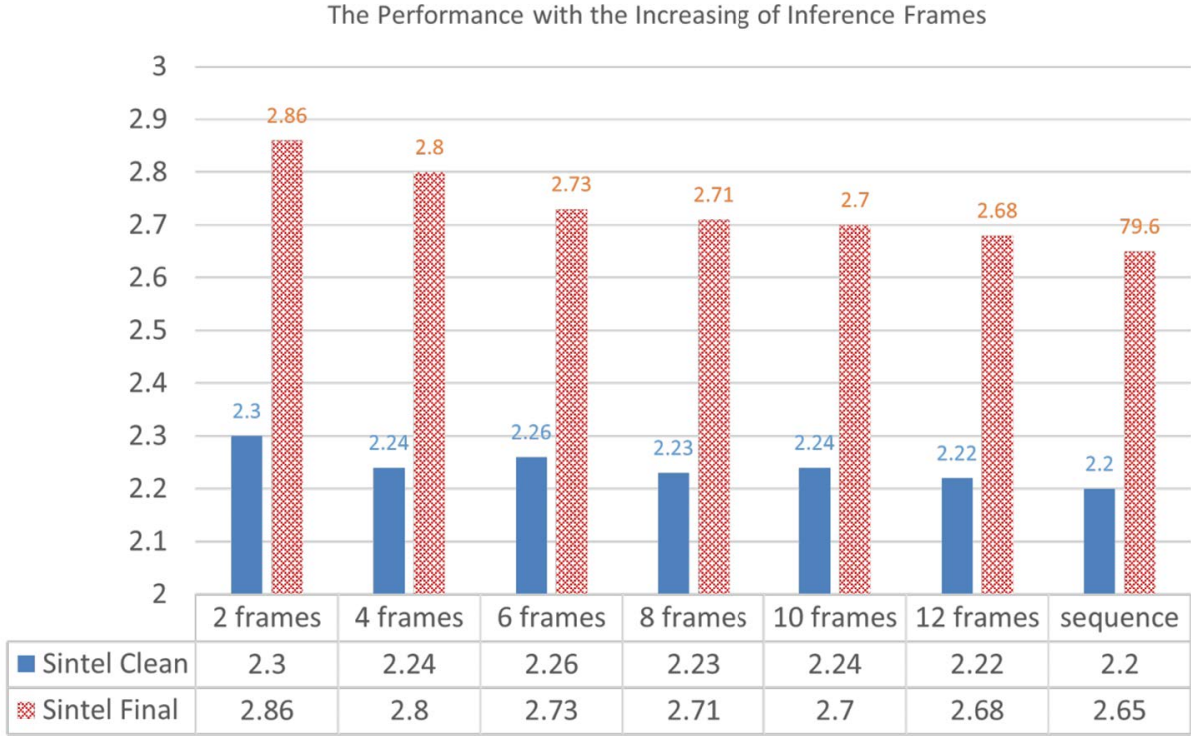


Figure 3.9: Network EPE performance with increasing numbers of inference frames. After multi-frame dynamic training, the model converges using sequences of arbitrary lengths. The sequence in the table indicates that the network uses all frames of the scene during inference. The model performance gradually improves as the number of inference frames increases.

working in complex and dynamic real-time environments.

We experimentally evaluated model runtime performance via inference over 100 consecutive frames (512×512). The real-time processing speed was only 17 ms per frame, which is more than two times faster than the well-known RAFT model (39 ms). During inference, our model recurrently iterates over new frames; as the number of frames increases, the processing time would be expected to increase linearly. However, recursive inference also produces more frames of optical flow as time passes, maintaining a constant average processing time per frame. This makes our model appropriate for processing continuous video streams, such as the Sintel dataset, without additional time. Our model is well-suited for dynamic real-time scenarios with over 50 FPS high inference speed.

Dynamic training enhancer

During self-supervised learning, the DTE improves the generalization of the three specific scenes and thus greatly enhances model performance. We separately validated the performances of the DOE, SVE, and CVE and combinations thereof.

Table 3.6 summarizes the DTE ablation studies using the Sintel Clean, Sintel Final, KITTI

Table 3.4: Ablation study on model architectures. The AEPEs in specific regions of the scene and the numbers of CNN parameters are shown.

Net Structure	Sintel Clean			Sintel Final			Param.
	ALL	NOC	OCC	ALL	NOC	OCC	
PWCNet [164]	2.48	1.19	21.71	3.47	1.98	25.19	8.75M
PWCNet-small [164]	2.76	1.28	23.92	3.62	2.16	28.15	4.05M
ARFlow [87]	2.30	1.08	20.00	3.19	1.84	22.77	2.24M
ARFlow-MV [87]	2.24	1.04	19.60	3.18	1.86	22.36	2.37M
Ours-seq	2.17	0.91	19.36	3.12	1.74	22.83	2.50M

Table 3.5: Ablation study using different combinations of the three training enhancers. Base: ULDENet featuring basic multi-frame unsupervised training. DOE: dynamic occlusion training enhancer; SVE: spatial variation training enhancer; CVE: content variation training enhancer.

Base DOE SVE CVE	Sintel Clean			Sintel Final			KITTI-15			KITTI-12		
	ALL	NOC	OCC	ALL	NOC	OCC	ALL	NOC	OCC	ALL	NOC	OCC
✓	2.17	0.91	19.36	3.12	1.74	22.86	2.65	1.92	5.86	1.42	0.92	4.32
✓ ✓	1.86	0.85	16.25	2.90	1.69	20.42	2.43	1.76	5.38	1.28	0.85	3.81
✓ ✓	1.96	0.75	17.87	2.98	1.63	22.17	2.55	1.82	5.77	1.35	0.86	4.18
✓ ✓ ✓	1.79	0.78	15.82	2.88	1.68	20.22	2.37	1.72	5.24	1.24	0.83	3.67
✓ ✓	1.92	0.79	17.23	2.94	1.66	21.36	2.50	1.86	5.60	1.31	0.86	3.96
✓ ✓ ✓ ✓	1.75	0.81	15.21	2.81	1.63	19.83	2.28	1.62	5.17	1.19	0.78	3.59

2012, and KITTI 2015 datasets, where the ‘Baseline’ is ULDENet with the basic unsupervised loss combination and TSR. Quantitatively, DOE alone reduces the EPE from 2.17 to 1.86, providing the best improvement of 14.3%, and the SVE and CVE yield 10% and 11.5% performance gains, respectively. Combining the three reduces the EPE from 2.17 to 1.75 (20% error reduction).

Upon separately evaluating the OCC and NOC regions, it’s evident that DOE most effectively refines predictions in the OCC zones, trimming the EPE from 19.36 to 16.25 for the Sintel Clean dataset. The CVE also slightly improved OCC regional accuracy; this can be attributed to the dynamic noise and regional blurring are similar to the information loss caused by occlusion. In the regular NOC regions, both the CVE and SVE reduced network prediction errors.

The dynamic variation and static image transformations are shown in Table 3.6, where CT and ST perform identical content & spatial image transformations across all frame sequences, including image flipping and Gaussian blurring similar to those of the data augmentation method

Table 3.6: Ablation study comparing dynamic scene variation to static data augmentation. Baseline: ULDENet structure with basic multi-frame unsupervised training; Stc Occ: static random noise masking; Dyc Occ-sparse / mix: dynamic occlusion simulation with mixed loss supervision. Stc CT / ST: static content transformation / spatial transformation. Dyc CV / SV: dynamic content variation / spatial variation.

Enhancer Strategy	Sintel Clean			Sintel Final		
	ALL	NOC	OCC	ALL	NOC	OCC
Baseline	2.17	0.91	19.36	3.12	1.74	22.83
+Stc Occ	2.06	0.90	18.21	3.03	1.78	21.15
+Dyc Occ	1.86	0.85	16.25	2.90	1.69	20.42
+ Stc CT	2.04	0.81	18.44	3.03	1.71	22.01
+ Dyc CV	1.92	0.79	17.23	2.94	1.66	21.36
+ Stc ST	2.07	0.77	19.00	3.08	1.73	22.44
+ Dyc SV	1.96	0.75	17.87	2.98	1.63	22.17

[87]. Stc Occ refers to image cropping and random noise occlusion of image subregions with the original unsupervised losses [90]. All hyperparameters were consistent with the dynamic scenario. It is clear that dynamic variation and occlusion achieved higher performance than static image augmentation; our dynamic occlusion simulation with mixed occlusion loss (Dyc Occ) outperforms Stc Occ by 10% gains.

3.5 Decoupled Spatiotemporal Fusion Network

The previous section 3.3 introduced a temporal dynamic scene argumentation method (Proposal I), demonstrating the effectiveness of considering temporal relations across frames. Building on Proposal I, this section modifies the inference structure of the spatiotemporal adaptive fusion network (STAFNet), which operates iteratively over a continuous sequence of frames to estimate the current optical flow.

Specifically, STAFNet’s function can be decoupled into two parts: the global correspondence matching (GCM) block and the spatiotemporal adaptive fusion (STAF) block. The GCM block is a time-independent model that focuses on the apparent correspondence between two images. In contrast, the STAF block is a temporal dynamic model that identifies and refines the LMC (Low Match Confidence) regions using spatial and temporal context information. By integrating the self-supervised learning method from Proposal I, the STAF block is expected to be driven to leverage spatiotemporal context to refine the LMC areas.

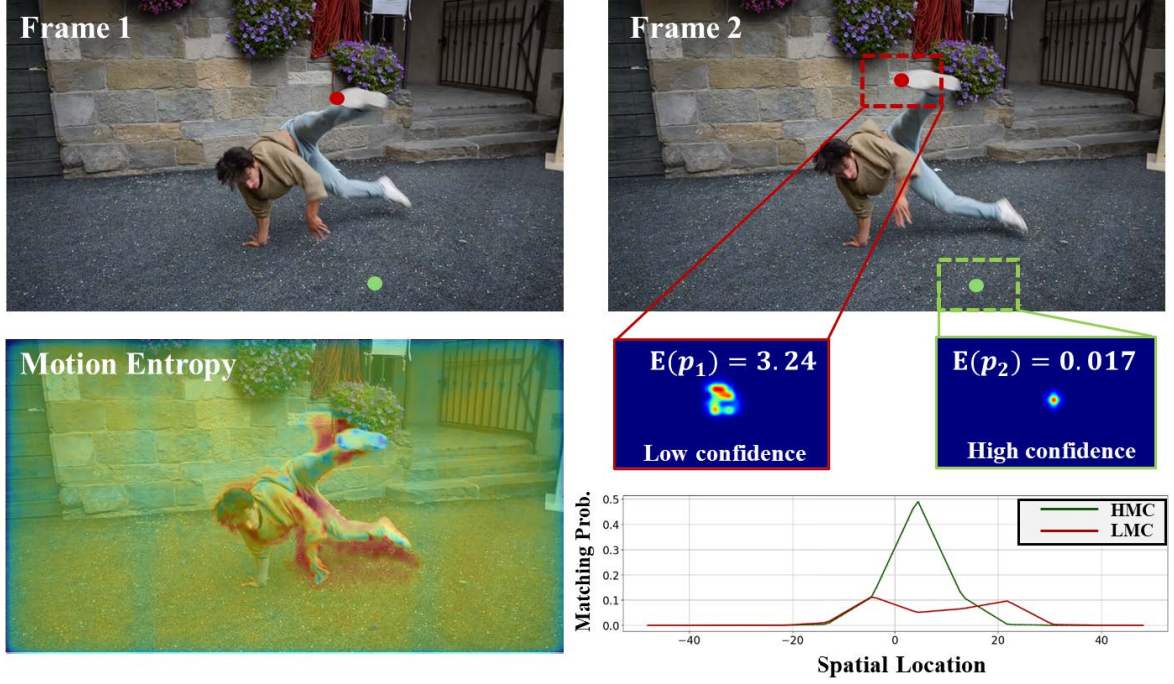


Figure 3.10: Correspondence entropy distribution: Each location’s correspondence can be considered as a two-dimensional probability distribution for which information entropy characterizes the degree of confidence in the matching. Figure is adapted from [168]

3.5.1 Global Correspondence Matching

The GCM block is designed to find apparent pixel correspondences between two images with the assumption that all pixels in a pair of images should have a unique correspondence.

Figure 3.11 shows two consecutive video frames $\{I_{t-1}, I_t\}$. We first extract downsampled dense features $\{X_{t-1}, X_t\} \in \mathbb{R}^{H \times W \times C}$ with a weight-sharing convolutional network, where H, W , and C denote height, width, and feature dimension, respectively.

Considering that the correspondences in the two frames should share high feature similarity, we first compare the feature similarity for each pixel in X_{t-1} with respect to all pixels in X_t by computing their inner product distance [61]. This can be implemented efficiently with simple matrix multiplication:

$$\mathbf{M} = \frac{X_{t-1} X_t^T}{\sqrt{C}} \in \mathbb{R}^{HW \times HW}, \quad (3.16)$$

where each scalar in the correlation volume $\mathbf{M}(x_1, y_1, x_2, y_2)$ represents the correlation value between a pair of coordinates (x_1, y_1) and (x_2, y_2) , and $\sqrt{C}$ is a normalization factor to avoid large values after the dot-product operation. For any location $p_1 = (x_1, y_1)$ in X_{t-1} , it is assumed that its corresponding location with the greatest similarity in X_t is $p_2 = (x_2, y_2)$. In this case p_1, p_2 should be considered as a correspondence between two images, thus the motion flow from p_1 to p_2 can be derived by subtracting the coordinates, i.e., $\mathbf{F}_{t-1 \rightarrow t}(p_1) = (x_2 - x_1, y_2 - y_1)$. Based

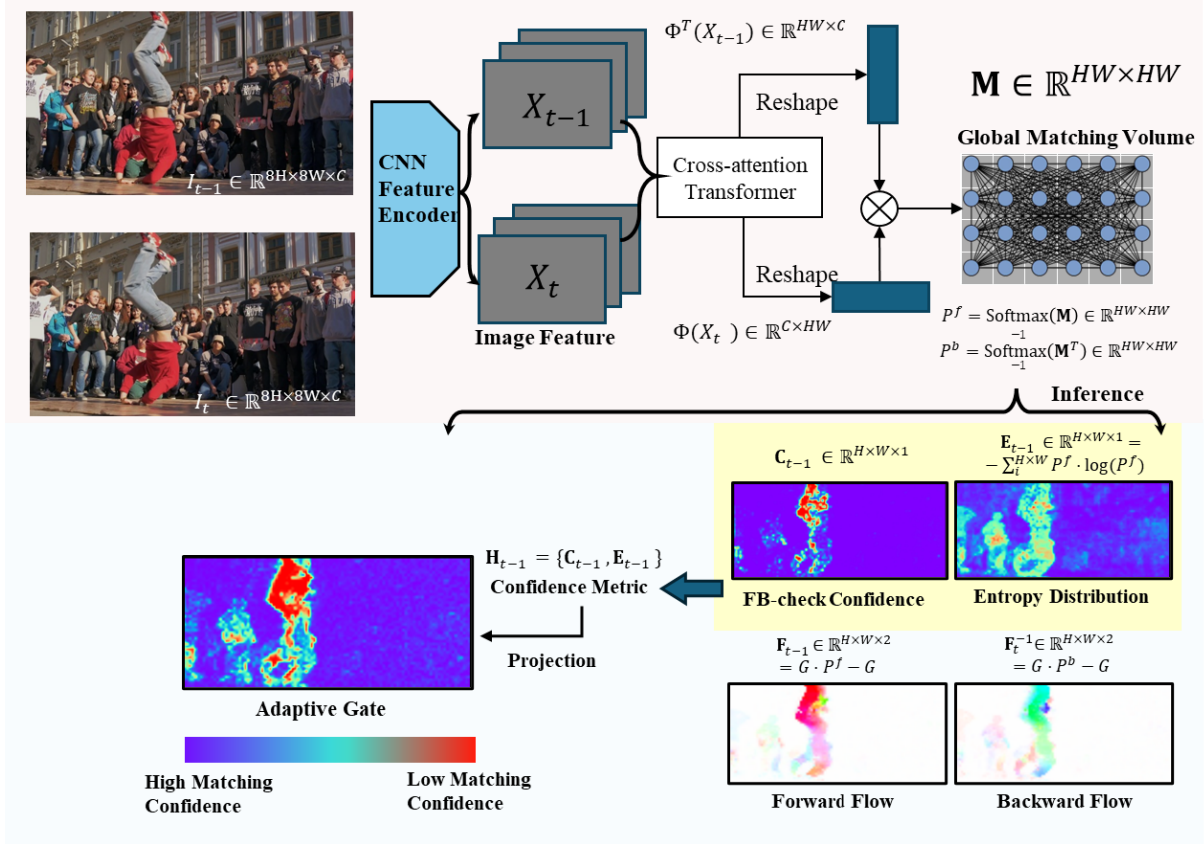


Figure 3.11: Global correspondence matching block (GCM): the GCM block uses only a pair of pictures to extract the global correspondence and the global correspondence volume is obtained by the inner product distance of the features. The coarse forward-backward optical flow, forward-backward consistency confidence, and entropy distribution can be inferred subsequently. The adaptive gate plays a vital role in controlling the fusion of spatiotemporal contextual information, the details of which are described in Fig. 3.12. Figure is adapted from [168]

on this principle, the dense motion flow from I_{t-1} to I_t can be calculated using the following equation:

$$\mathbf{F}_{t-1 \rightarrow t} \in \mathbb{R}^{H \times W \times 2} = \text{softmax}(\mathbf{M}) \cdot \mathbf{G} - \mathbf{G} \quad (3.17)$$

where $\mathbf{G} \in \mathbb{R}^{HW \times 2}$ are the normalized 2D Euclidean coordinates of whole images. Softmax is a flexible way to assign the maximum correspondence of each spatial coordinate differentially.

We adopt the structure of the supervised learning method, GMFlow [187], which uses the Swin-transformer [92] to search global corresponding matching between two images. Specifically, six layers of stacked transformer blocks are used to extract deeper features based on attention mechanisms. We define the transformer operation between $\{X_{t-1}, X_t\}$ as $\Phi_t^{t-1}(\cdot) : \mathbb{R}^{H \times W \times C} \mapsto \mathbb{R}^{H \times W \times C}$. The embedded features $\{\Phi_t^{t-1}(X_{t-1}), \Phi_t^{t-1}(X_t)\}$ are used for global correlation matching to derive the dense motion flow, as represented in Eqs. 3.16, and 3.17.

The main difference in the transformer part with GMFlow is that calculations are performed on the 1/8 scale, which drastically reduces the computational complexity, and reduces the inference time by 30% of the original time. Furthermore, reducing the window size of the self-attention block to 1/2 of the original setting does not result in any significant performance drop, but the computational complexity is reduced by half.

Matching Confidence Metric

The above approach works well for objects with clear correspondence between pairs of images, but it cannot effectively address pixels without an apparent matching relationship. For example, if an object in I_{t-1} is occluded or blurred in I_t , the matching relationship would be ambiguous. We propose two strategies to evaluate the confidence in global matching that provide essential guidance for the subsequent adaptive fusion block.

As shown in Figure 3.10, we first analyze this phenomenon by visualizing the entropy distribution of the correspondence volume. According to Eq. 3.16, we denote the normalized correspondence volume as P , i.e., $P = \text{Softmax}(\mathbf{M}) \in \mathbb{R}^{HW \times HW}$. We assume that each location (i, j) in the first frame has a potential corresponding location in the second frame, which can be represented by a two-dimensional probability distribution $P(i, j) \in \mathbb{R}^{H \times W}$. When matching correspondence at (i, j) becomes ambiguous due to information loss, the two-dimensional probability distribution correspondingly becomes divergent, resulting in a high entropy value indicating that it is difficult to solve a clear correspondence. According Shannon's concept of entropy [150], the correspondence entropy of any location (i, j) can be obtained, as follows:

$$\mathbf{E}(i, j) = - \sum_{i=0}^H \sum_{j=0}^W P(i, j) \log P(i, j) \quad (3.18)$$

Another confidence metric is based on the forward-backward optical flow consistency check. First, one can obtain the backward motion flow by performing the inverse operation of Eq. 3.17., i.e.,

$$\mathbf{F}_{t \rightarrow t-1} \in \mathbb{R}^{H \times W \times 2} = \text{Softmax}(\mathbf{M}^T) \cdot \mathbf{G} - \mathbf{G}. \quad (3.19)$$

For a moment t , the forward flow $\mathbf{F}_{t-1}$ and backward flow $\mathbf{F}_t^{-1}$ should satisfy: $\mathbf{F}_{t-1} = -\mathbf{F}_t^{-1}(\mathbf{p} + \mathbf{F}_{t-1}(\mathbf{p}))$. This equation does not hold in regions where the correspondence between the forward and backward directions does not match, thus defined as a low-confidence region (i.e., unmatched region). Based on this definition, the confidence $\mathbf{C} \in \mathbb{R}^{H \times W \times 1}$ can be described by:

$$\mathbf{C} = \text{Tanh} \left(\frac{\left| \mathbf{F}_{t-1} + \mathbf{F}_t^{-1}(\mathbf{p} + \mathbf{F}_{t-1}(\mathbf{p})) \right|^2}{(|\mathbf{F}_{t-1}|^2 + |\mathbf{F}_t^{-1}|^2) \times \delta} \times \alpha \right). \quad (3.20)$$

The denominator is the weight decay based on the optical flow displacement; $\delta = 0.01$ is a fixed scale factor, $\text{Tanh}(\cdot)$ is a nonlinear mapping function that compresses the confidence level between $(0, 1)$; $\alpha > 0$ is a trainable scale factor controlling the shape of the mapping function.

Both entropy distribution and forward-backward consistency are used to represent the matching confidence, as $\mathbf{H} = \{\mathbf{E}, \mathbf{C}\}$, which are further transformed into an adaptive gate via nonlinear projection. The adaptive gate plays a vital role in controlling the fusion of spatiotemporal contextual information, thereby providing additional temporal and spatial context to regions with low confidence. The visualization is shown in Figure 3.11.

3.5.2 Spatial-Temporal Adaptive Fusion

For regions with no obvious matching relationship, an effective approach is to optimize the motion flow using more historical motion-context and spatial-context information. It is worth noting that for the HMC region, only two frames are sufficient to find an accurate corresponding match, and fusing additional information would instead make the model struggle to converge due to information redundancy. Therefore the designed spatiotemporal adaptive fusion module uses the redundant spatiotemporal contextual information in the LMC region.

Temporal Recurrent Fusion

The inference pipeline of STAFNet over continuous causal sequences is shown in Figure 3.12, where the GCM block in the left part of subfigure (A) is used to generate explicit correspondences and coarse forward-backward optical flows. The confidence measure of the correspondence is inferred simultaneously, as expressed by the entropy distribution and the consistency of forward-backward flows. The motion flow, image features, and correspondence confidence are first non-linearly embedded (1×1 convolution with Tanh activation) as a 128-dimensional spatial motion context $\mathcal{S} \in \mathbb{R}^{H \times W \times 128}$, which is subsequently fused with the motion Gated Recurrent Unit (GRU) to obtain temporal contextual feature $\mathcal{T} \in \mathbb{R}^{H \times W \times 128}$.

The GRU is used to integrate the motion context at each time step and to provide a long-term historical motion context for the current temporal state, which is helpful for the network to predict the motion of the LMC region. Basically, motion GRU follows the principle of the original concept of GRU [23] but replaces the operation with fully convolutional approaches; it takes the previous temporal context $\mathcal{T}_{t-2}$ and current motion feature $\mathcal{S}_{t-1}$ as input and generates a fused feature $\mathcal{T}_{new}$ that serves as the temporal motion context, the process of which is formalized as:

$$\begin{aligned}
 \hat{\mathcal{T}} &= \mathcal{W}_{\mathbf{F}_{t-1}^{-1}}(\mathcal{T}_{t-2}) \\
 z &= \sigma \left(\text{Conv}_{3 \times 3} \left(\text{ConCat} \left[\hat{\mathcal{T}}, \mathcal{S}_{t-1} \right] \right) \right) \\
 r &= \sigma \left(\text{Conv}_{3 \times 3} \left(\text{ConCat} \left[\hat{\mathcal{T}}, \mathcal{S}_{t-1} \right] \right) \right) \\
 q &= \text{Tanh} \left(\text{Conv}_{3 \times 3} (\text{ConCat}[r \odot \mathcal{T}_{t-2}, \mathcal{S}_{t-1}]) \right) \\
 \mathcal{T}_{new} &= (1 - z) \odot \mathcal{S}_{t-1} + z \odot q.
 \end{aligned} \tag{3.21}$$

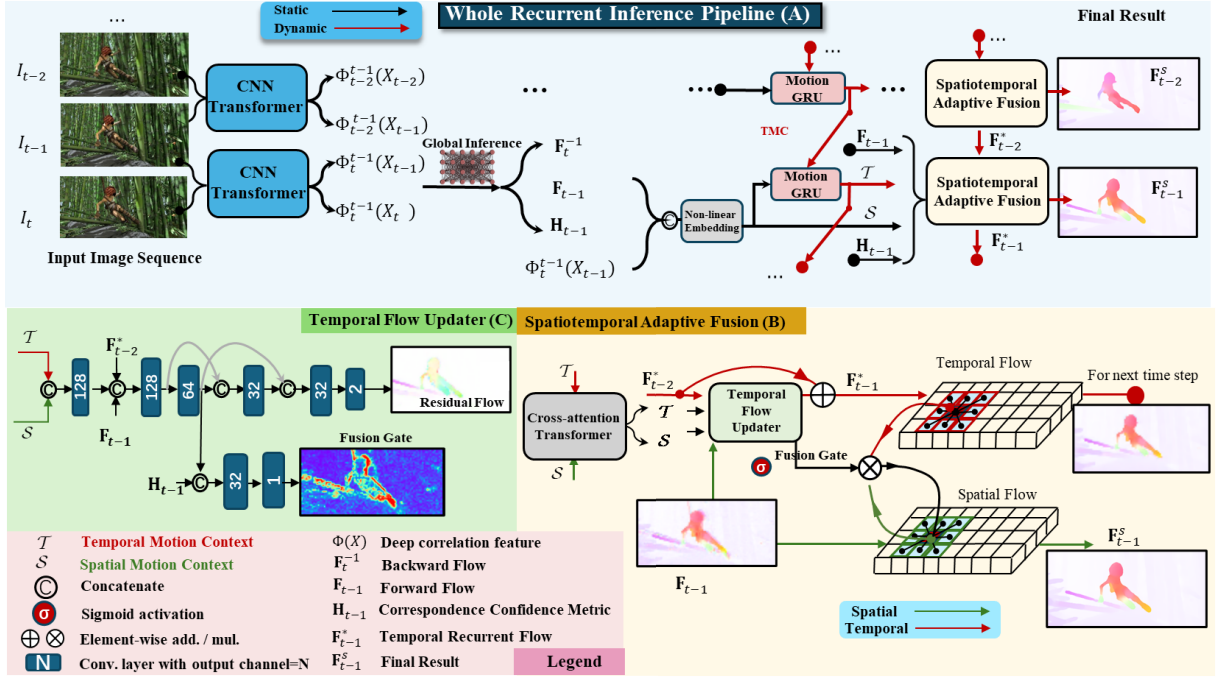


Figure 3.12: Whole inference pipeline of the spatiotemporal adaptive fusion network (STAFNet): the GCM Block, illustrated in Figure 3.11 and shown in the darker blue part of subfigure (A), is a temporal static model that relies only on the current two images to reason through the apparent forward-backward flows, which are subsequently fed into the STAF block for further refinement, as illustrated in subfigure (B). The STAF block re-predicts the motion field of the low-matching-confidence (LMC) regions by tracking the more spatiotemporal motion context. Figure is adapted from [168]

where $\sigma(\cdot)$ denotes a sigmoid function; $\text{ConCat}(\cdot)$ denotes the concatenation of feature maps; and $\odot$ presents element-wise production. At each GRU fusion stage, the temporal motion context of the previous moment is spatially aligned with the current moment by warping the operation of backward optical flow $\mathcal{W}_{F_{t-1}^{-1}}(\cdot)$, and the whole process is differentiable.

After the GRU, the current spatial motion context and the historical temporal motion context are generated, both of which are fed into the adaptive fusion block to optimize the LMC regions.

Adaptive Fusion Block

Due to information loss, there is no direct correspondence in LMC regions. Predictions of LMC regions can be rescued by considering additional temporal and spatial contextual relationships of the motion field. Spatial smoothness is widely present in natural objects, and is accompanied by similar optical flows in adjacent locations. It is no coincidence that the identical object also obeys the smoothness principle in adjacent time steps due to the restrictions of physical laws, resulting in similar motion fields.

The STAF block is illustrated in Figure 3.12 (B). After applying an additional layer of the cross-attention transformer, the spatial and temporal contexts are fed into the temporal flow updater. As illustrated in sub-figure (C), the temporal flow updater contains two output pathways that act on different functionalities; specifically to update the temporal flow $\mathbf{F}^*$ via recurrent residual compensation and to transform confidence $\mathbf{H}$ to the adaptive fusion gate $\mathcal{G} \in \mathbb{R}^{H \times W}$.

The temporal flow is computed using the historical motion context and the current motion context, which is recurrently updated over time. We formulate the first path of the flow updater as a nonlinear mapping to generate residual flow $u_\theta : \mathbb{R}^{H \times W \times C} \mapsto \mathbb{R}^{H \times W \times 2}$. The updating law of the temporal flow is represented by the following:

$$\mathbf{F}_t^* = \mathcal{W}_{\mathbf{F}_t^{-1}}(\mathbf{F}_{t-1}^*) + u_\theta \left(\mathcal{W}_{\mathbf{F}_t^{-1}}(\mathbf{F}_{t-1}^*), \mathbf{F}_t, \mathcal{T}, \mathcal{S} \right). \quad (3.22)$$

The warping operation $\mathcal{W}_{\mathbf{F}_t^{-1}}(\cdot)$ is to track the object's location at the previous time step by backward flow and warp the grids to the current location. Considering that there is also a tiny variation in the instantaneous motion of the identical object within the adjacent frames, the second term is attempts to this variation and rectify it using residual compensation.

For each LMC location $\mathbf{p}' = (i, j)$, we define a proximity domain around p

$$\mathcal{N}(\mathbf{p}')_r = \{\mathbf{p}' + \Delta x | \Delta x \in \mathbb{Z}^2, 0 < |\Delta x| \leq r\} \quad (3.23)$$

as the set of integer offsets that are within a radius of r units of $\mathbf{p}'$ using the L1 distance. Considering the strong motion similarity in proximal locations, all of the flow vectors in spatial flow $\{\mathbf{F}_{t-1}(p) | p \in \mathcal{N}(\mathbf{p}')_r\}$ and temporal flow $\{\mathbf{F}_{t-1}^*(p) | p \in \mathcal{N}(\mathbf{p}')_r\}$ are weighted and integrated to predict a new flow vector. We use an attention mechanism to generate fusion weights, i.e., the weight of spatial fusion depends on the inner product distance of features between the spatial motion context $\mathcal{S}(\mathbf{p}') \in \mathbb{R}^{128}$ and each adjacent location $\mathcal{N}(\mathbf{p}')_r$ in $\mathcal{S}$, whereas temporal fusion weights depend on the inner product distance between $\mathcal{S}(\mathbf{p}')$ and each location $\mathcal{N}(\mathbf{p}')_r$ in the temporal context $\mathcal{T}$. The specific operation is expressed as

$$\begin{aligned} \mathbf{W}_{ss} &= \left\{ \frac{\mathcal{S}(\mathbf{p}') \cdot \mathcal{S}^T(p)}{\sqrt{C}} \mid p \in \mathcal{N}(\mathbf{p}')_r \right\} \\ \mathbf{W}_{st} &= \left\{ \frac{\mathcal{S}(\mathbf{p}') \cdot \mathcal{T}^T(p)}{\sqrt{C}} \mid p \in \mathcal{N}(\mathbf{p}')_r \right\} \\ \mathbf{W}_{ss}, \mathbf{W}_{st} &= \text{Softmax}(\text{ConCat}[\mathbf{W}_{ss}, \mathbf{W}_{st}]), \end{aligned} \quad (3.24)$$

where the $\text{Softmax}(\cdot)$ is used for energy normalization.

Eventually, the refined optical flow is generated by the following equation, where $\mathcal{G}$ serves as a gating adaptive control of fusion level for different locations, thus assigning more spatiotemporal context information to the LCM regions.

$$\begin{aligned} \mathbf{F}_{t-1}^s(\mathbf{p}') &= \sum_{p \in \mathcal{N}(\mathbf{p}')_r} \mathcal{G}(p) \cdot \mathbf{W}_{ss}(p) \cdot \mathbf{F}_{t-1}(p) + \\ &\quad \mathcal{G}(p) \cdot \mathbf{W}_{st}(p) \cdot \mathbf{F}_{t-1}^*(p) + (1 - \mathcal{G})\mathbf{F}_{t-1}(p) \end{aligned} \quad (3.25)$$

3.5.3 Decoupled Training

Training strategy is an important aspect of unsupervised learning tasks. We innovatively decoupled the optimization objective to fine-tune the GCM and STAF blocks, respectively. The GCM block is focused on extracting the apparent correspondence between two images, whereas the STAF module learns to explore more spatiotemporal contextual relationships to optimize the LMC regions.

Basic Unsupervised Training

The basic unsupervised learning strategy is employed for the global correspondence block. We follow similar strategies in section 3.3.3 that sequentially feed N consecutive frames into the network and compute the loss of generated $N - 1$ optical flows for backward propagation.

Specifically, at the end of a sequence, we update the weights of the GCM block so as to decrease:

$$\mathcal{L}_{GCM} = \frac{1}{N-1} \sum_{t=1}^{N-1} \mathcal{L}_t \quad (3.26)$$

where $\mathcal{L}_t$ is the combination loss for image pair (I_t, I_{t+1}) .

Optimizing LMC regions via self-supervised training

There is a variety of variants in natural video sequences that result in ambiguous correspondence between adjacent frames, including mutual occlusion within objects, objects shifting outside image boundaries, motion blur caused by rapid movement, out-of-focus blur caused by camera perspective shifts, the illusion of light and shadow caused by variation in local/global illumination conditions, etc.

To enable the STAF block to cope better with these situations, we coupled our proposal I, which simulates various challenging scenes and utilizes the self-supervised learning approach for self-distillation.

Simulation of LMC regions

We incorporated the dynamic scene augmentation method in section 3.3.4 and simulated various possible scenarios of LMC regions in continuous sequences. In particular, as the STAF model is a dynamic temporal block that extracts historical motion information over a long period, we follow the laws of physics in our simulations, i.e., continuity and smoothness of motion and variation in natural environments. Such laws are crucial for the model to exploit spatiotemporal contextual information. For example, for a current object A, it may be difficult to obtain an accurate correspondence due to dynamic out-of-focus blurring and occlusion. However, the object is clearly visible in the previous time frame (which is very common in a video sequence). Thus the temporal motion smoothness is valuable for predicting the current trajectory of A.

Occlusion is one of the most common phenomena that leads to Loss of Motion Consistency (LMC). We modified the previous mixed supervision method described in Section 3.3.4 into a more direct self-supervised approach. This involves generating pseudo labels for artificial occluders using inverse affine transformation. Specifically, the affine transformations on an image denote the shift of the spatial coordinates of pixels, which act on the whole sequence, as represented by:

$$\{\tilde{I}_t(\mathbf{p})\}_{t=1}^N = \{I_t(\tau_{\theta_t}(\mathbf{p}))\}_{t=1}^N, \quad (3.27)$$

where $\tau_{\theta_t}(\cdot)$ is the transformation of pixel coordinates at image I_t (the occluder in this case) with transform parameters of θ_t . As different transformation parameters are applied on I_t and I_{t+1} will lead to variation in the optical flow field, causing the illusion of apparent motion. In this case, one should track the flow variation between $\{\tau_{\theta_t}, \tau_{\theta_{t+1}}\}$ via inverse-affine transformation, so as to generate forward optical flow of occluders $\mathbf{V}^\delta$, i.e.,

$$\{\mathbf{V}_t^\delta(\mathbf{p})\}_{t=1}^{N-1} = \{\tau_{\theta_{t+1}}^{-1}(\mathbf{p}) - \tau_{\theta_t}^{-1}(\mathbf{p})\}_{t=1}^{N-1}.$$

Eventually, the flow of the occluders will directly replace the original object flow at its spatial location. Defining the spatial grid covered by the occluder A as Φ_A , the generation process of pseudo label $\hat{\mathbf{V}}$ is given by:

$$\{\hat{\mathbf{V}}_t(\mathbf{p})\}_{t=1}^{N-1} = \begin{cases} \{\mathbf{V}_t(\mathbf{p})\}_{t=1}^{N-1}, & \mathbf{p} \notin \Phi_A \\ \{\mathbf{V}_t^\delta(\mathbf{p})\}_{t=1}^{N-1}, & \mathbf{p} \in \Phi_A \end{cases}$$

In the experiment, multiple occluders were created with independent motion trajectories. During occluder movement, the original object is smoothly obscured across multiple frames, naturally creating LMC regions that can be supervised using pseudo-labels, as shown in Figure 3.13.

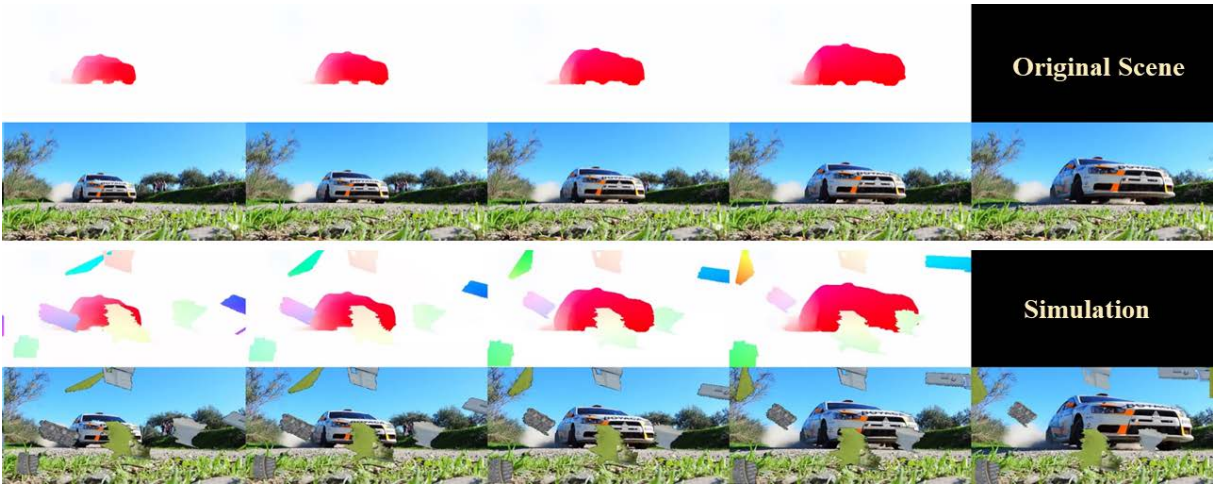


Figure 3.13: Simulation of dynamic occlusion and with the pseudo label. Figure is adapted from [169].

3.5.4 Efficient Training Step

The basic unsupervised training in subsection and the self-supervised training in subsection 3.5.3 are combined to optimize the parameters of GCM and STAF blocks. During the inference, the current two frames are first fed into the GCM, and the apparent correspondence volume is obtained via Eq. 3.16. Through further inference, one can obtain the forward flow, backward flow, forward-backward consistency confidence, and entropy distribution simultaneously, where the forward flow is supervised by Eq. 3.26.

The coarse optical flow generated by the GCM block is optimized in the STAF block using spatial-temporal contexts and by developing the final optimized result, as well as the intermediate product, recurrent temporal flow, both of which are optimized using self-supervised loss functions from Eq. 3.7. The whole network parameters are supervised by

$$\mathcal{L}_{all} = \frac{1}{N-1} \sum_{t=1}^{N-1} \lambda_1 \ell_{ph} + \lambda_2 \ell_{sm} + \lambda_3 \ell_{self}. \quad (3.28)$$

Due to limitations of GPU memory, images are always cropped to be smaller in training (e.g., images in the Sintel dataset are typically cropped from 436×1024 to less than 384×832). Likewise, the number of frames input per inference is restricted because the memory burden increases linearly with the number of frames.

In practice, we optimize the training process to reduce the GPU storage burden by disabling the computational graph cache, which is easily implemented in the PyTorch framework with the `torch.no_grad()`. In the experiments, the accuracy of the inference results was proportional to the number of input frames. Similarly, a larger image view effectively reduced the boundary occlusion effect. Therefore, we designed an **E**fficient self-supervised **T**raining algorithm (ET) based on the temporal dynamic model to take advantage of larger-view images and longer temporal sequences.

Specifically, we first perform forward and backward inference using images at their original size in relatively long sequences and obtain the forward and backward flows separately. High-confidence results are extracted by discarding the initial results of the front and tail frames, which serve as pseudo-labels for subsequent self-supervision. The non-cropped results are also used to generate a full-view occlusion mask $\mathbf{O}$, accompanied by boundary dilated warp [91], to perform global warping. The above process does not consume additional GPU storage, because the gradient tracking is disabled. In addition, we carry out backpropagation and empty the cache immediately after the completion of each subtask. We do not sum the loss functions before updating the weights uniformly. This strategy is similar to that of multitasking optimization, which significantly reduces the memory load while preserving performance [194]. The specific algorithm is listed in Algorithm 1 for a more detailed description.

In the experiments, in addition to reducing the GPU storage burden over by 70% compared to the conventional self-supervised approach [87], the proposed training algorithm also significantly improved the model’s performance, as shown in Table 3.7.

Algorithm 1 Efficient self-supervised Training algorithm (ET)

Input: Dataset containing consecutive frames $\{I_t\}_{t=1}^\infty, s.t. I_t \in \mathbb{R}^{H \times W \times 3}$, model parameters

$$\Theta = \{\Theta_{gcm}, \Theta_{staf}\}$$

1: Initialization: $\Theta_{gcm}, \Theta_{staf}, N, M, K, n = 0$

2: **repeat**

3: $n \leftarrow n + 1$

4: loading N consecutive frames $\mathcal{I} = \{I_t\}_{t=1}^N$

5: Stop GPU caching of computation graph (Inference mode)

6: Computing forward flow: $\{V_t\}_{t=1}^{N-1} = f_\Theta(\mathcal{I})$

7: Computing backward flow: $\{V_t^{-1}\}_{t=N}^2 = f_\Theta(\mathcal{I}^{-1})$

8: Inferring full-view occlusion mask $\{O_t\}_{t=1}^{N-1}$ by consistency check

9: Cropping the sequence to a smaller size; removing the first and last K images in the sequence:

$$\{\hat{I}_t\}_{t=1}^{N-2K} \leftarrow \{I_t\}_{t=1}^N; \{\hat{V}_t\}_{t=1}^{N-2K-1} \leftarrow \{V_t\}_{t=1}^{N-1}; \{\hat{O}_t\}_{t=1}^{N-2K-1} \leftarrow \{O_t\}_{t=1}^{N-1}$$

$$\mathbf{I}, \mathbf{V}, \mathbf{O} = \{\hat{I}_t\}_{t=1}^{N-2K}, \{\hat{V}_t\}_{t=1}^{N-2K}, \{\hat{V}_t\}_{t=1}^{N-2K}$$

10: Enable GPU caching of computation graph (Training mode)

11: Feeding the $\mathbf{I}$ into model, the loss is calculated based on E.q. 3.26, and $\mathbf{O}$ is used to avoid low-confidence regions

12: Backpropagation and updating Θ_{gcm} ; Free GPU cache

13: **repeat**

14: Utilizing $\mathbf{I}, \mathbf{V}$ to create new a scene with LMC regions $\hat{\mathbf{I}}$ and corresponding pseudo labels $\hat{\mathbf{V}}$ based on Section 3.5.3

15: Feeding the $\mathbf{I}$ into the model; the loss is calculated based on Eq. 3.7, and $\hat{\mathbf{V}}$ is used for self-supervision

16: Backpropagation and updating Θ ; free GPU cache

17: **until** Finishing all LMC scenarios

18: **until** $n > M$

Output: Optimized model parameters Θ^*

Table 3.7: Main ablation study on different combinations of the proposed components. GCM: Global Correspondence Matching block baseline; STAF: Spatial-Temporal Adaptive Fusion block; UT: Unsupervised Training; DT: Decoupled Training; ET: Efficient Training algorithm based on Decoupled Training

GCM	STAF	UT	DT	ET	Sintel Clean			Sintel Final			KITTI			Param.	R.	T.
					ALL	NOC	OCC	ALL	NOC	OCC	ALL	NOC	OCC			
✓			✓		2.31	0.88	21.07	3.34	1.72	26.61	2.16	1.17	7.72	4.64 M	22 ms	
✓	✓		✓		2.04	0.81	18.44	2.97	1.68	21.48	1.87	1.06	6.43	7.04 M	28 ms	
✓	✓			✓	1.48	0.73	12.63	2.22	1.30	15.52	1.40	0.94	4.02	7.04 M	28 ms	
✓	✓		✓	✓	1.32	0.69	11.06	2.01	1.24	13.30	1.33	0.92	3.67	7.04 M	28 ms	

3.6 Experiment II

3.6.1 Implementation Details

The hyperparameter settings for the loss configuration, optimizer, and GPU machine environment in Experiment II were similar to those used in Experiment I (Section 3.4). We first pre-trained the model on FlyingChair with batch sizes of 8 and a learning rate of $2e^{-4}$, followed by two stages of formal training on the Sintel or KITTI dataset. In the first stage, we used the basic unsupervised loss function with a sequence length of 6, a batch size of 4, and a learning rate of $1e^{-4}$ for 100 epochs of training. The self-supervision weight was increased linearly from 0 to our setting. In the second stage, we performed 50 epochs of intensive training using self-supervised distillation, in which the sequence length was set to 12 per iteration ($N = 12$, $K=2$ in Algorithm 1).

For image preprocessing, the images from the Sintel and DAVIS datasets were first resized to 448×1024 then cropped to 384×832 and inferred at the original size in the validation; the images in the KITTI dataset were resized to 384×1216 , and then cropped to 256×832 for training and 384×1216 for validation. We also adopted random image horizontal flipping and sequence reversion for dataset augmentation.

In training, we mixed Clean, Final, and Albedo scenes from the Sintel training set and paired them for training. As for training on the KITTI dataset, we employed its multi-view extension version for sequence training. Notably, we only touched the images of the training set in all training procedures, and the flow ground truth was only used for validation.

3.6.2 Ablation Study

An extensive ablation study was conducted to demonstrate the effectiveness of each technical component. The Sintel training set was subdivided into separate training, and validation sets, following the settings in [87].

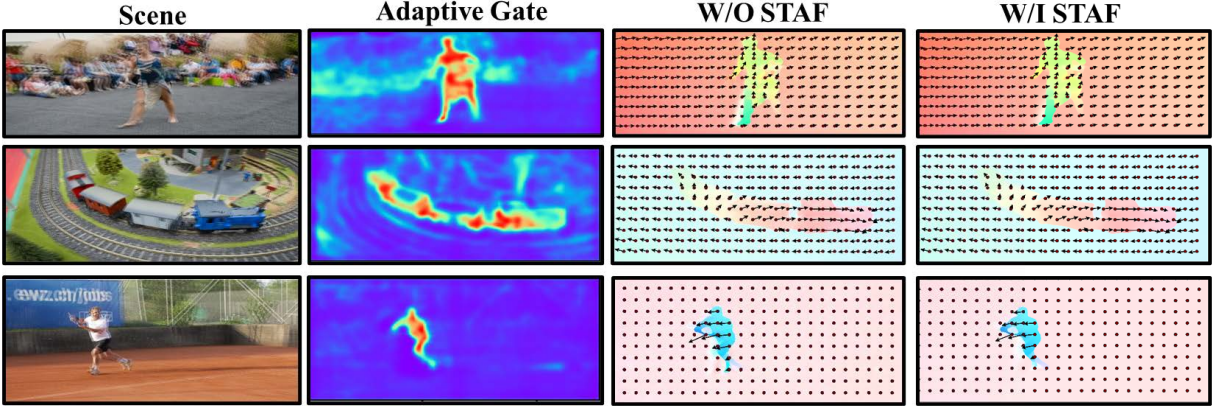


Figure 3.14: Quantitative demonstration on DAVIS dataset. Figure is adapted from [168].

Global Evaluation

We validated the proposed blocks globally, including GCM, STAF, decoupled training, and ET algorithms. The quantitative data are described in Table 3.7, where the occlusion and non-occlusion regions of the Sintel and KITTI datasets are reported. From the data, the proposed components all improved the network performance independently. The STAF block did not increase the inference time excessively, although it introduced an additional 2.4 M parametric count. The combination of the decoupled training and the STAF block cumulatively reduced the EPE from 2.31 to 1.48 (a 36% performance improvement) for the Sintel Clean dataset, and from 2.16 to 1.40 (a 35% performance improvement) for the KITTI dataset. Combined with the well-designed ET algorithm, the final performance improvement was close to 43% for the Clean dataset and 38% for the KITTI datasets.

Evaluation on Confidence Metric

We employed entropy distribution $\mathbf{E}$ and forward-backward consistency confidence $\mathbf{C}$ to characterize the LMC regions, which were mapped as the core metric to guide the fusion of spatiotemporal contexts by adaptive gating. We verified the effectiveness of $\mathbf{E}$, $\mathbf{C}$, and Adaptive Gating, respectively; the results are shown in Table 3.8. The Baseline in the first row of the table was trained using the standard unsupervised approach, and then we fixed the value of each confidence metric to one and retrained the model to convergence to obtain the data. Qualitatively, adding gating provides flexible control of the spatiotemporal context flow into the LMC regions. Furthermore, the entropy distribution and consistency confidence provide more useful clues for the evaluation of LMC regions. The final combined strategy improved the network performance by about 13%.

Table 3.8: Ablation study on LMC metrics. Baseline: STAFNet with unsupervised training; σ : Adaptive Gating Control; E: Entropy distribution; C: Forward-backward consistency confidence.

σ	E	C	Sintel Clean			Sintel Final		
			ALL	NOC	OCC	ALL	NOC	OCC
<i>Baseline</i>			2.34	0.90	21.30	3.33	1.80	26.55
✓	-	-	2.25	0.84	20.61	3.18	1.75	23.60
✓	✓	-	2.09	0.86	18.73	3.01	1.76	21.10
✓	-	✓	2.17	0.82	19.83	3.08	1.72	22.55
✓	✓	✓	2.04	0.81	18.84	2.97	1.68	21.48

Table 3.9: Ablation study of LMC augmentation. Baseline: STAFNet with unsupervised training; Pseudo-label: Occlusion simulation by pseudo-labeling; All: Combined three augmentations together.

Simulation Scene	Sintel Clean			Sintel Final		
	ALL	NOC	OCC	ALL	NOC	OCC
Baseline	2.04	0.81	18.84	2.97	1.68	21.48
+ pseudo-label	1.64	0.79	14.10	2.49	1.62	15.45
+ regional blur	1.76	0.76	15.59	2.62	1.57	17.88
+ color variation	1.80	0.74	16.14	2.73	1.48	20.52
+ All	1.48	0.73	12.63	2.22	1.30	15.52

Evaluation of LMC region simulation

In the proposed decoupled training, we designed three scenarios causing LMC, including occlusion pseudo labeling, random local blurring, and global/local illumination variation. The quantitative statistics are presented in Table 3.9. All three simulated scenarios improved the model’s performance, with masking pseudo-labeling improving the performance most significantly.

3.6.3 Comparison with State-of-the-art

We compared our approach with several current leading methods using both the KITTI and Sintel datasets. STAFNet achieves the best results on both the test and training sets, with a considerable margin on the Sintel dataset. The quantitative comparisons are presented in Table 3.10, where we simultaneously compare the proposed method with the current advanced supervised and unsupervised methods. From the EPE measure, SAFTNet’s results are among the top in all metrics in the Sintel dataset and ahead of SMURF [159] by 22% accuracy for

Table 3.10: Comparison with previous methods on the Sintel Benchmark. We used the average EPE error (the lower the better) as the evaluation metric for the Sintel datasets. Missing entries ‘-’ indicate that the result is not reported in the compared paper, and (·) indicates that the testing images were used during unsupervised training. The best unsupervised and supervised results are **bolded**.

Method		Sintel Train		Sintel Test		Param	RT
		Clean	Final	Clean	Final		
Supervised	RAFT [171]	(0.77)	(1.20)	2.08	3.41	5.26 M	39 ms
	RAFT-OOD	4.44	5.97	-	-	5.26 M	39 ms
unsupervised	UPFlow [99]	(2.33)	(2.67)	5.24	6.50	3.49 M	30 ms
	OIFlow [91]	(2.44)	(3.35)	4.26	5.71	5.26 M	-
	SMURF [159]	(1.71)	(2.58)	3.15	4.18	5.26 M	39 ms
	ULDENet (Exp. I)	2.20	2.55	3.50	4.70	2.50M	30 ms
	SAFTNet	(1.20)	(1.60)	2.44	3.89	7.04 M	28 ms
	SAFTNet-DAVIS	(1.29)	(1.76)	2.42	3.71	7.04 M	28 ms

Clean and 7% for Final on the test set. In addition, STAFNet is even ahead of many of the supervised learning models [51, 66, 164], but still has a considerable gap with the solid supervised learning model RAFT [171]. However, the disadvantage of supervised learning is that the dataset is restricted to specific domains, and cross-domain data distribution usually leads to severe performance degradation. We selected RAFT for verification. The results are shown in Table 3.10, RAFT-OOD (out of domain), demonstrates that the performance of training with a large amount of out-of-domain data was still not as good as unsupervised learning within the domain, where the RAFT-OOD was trained using a large sample combination out of the domain, including data from KITTI, Flythings, Flychairs, etc., and was fine-tuned on the KITTI dataset.

When considering the inference speed, network parameters, and accuracy, despite the additional parameter burden, we maintained the advantage of high-speed inference while achieving the best accuracy, which is more critical because several megabytes of parameters are not a problem for today’s storage devices, but the speed of inference affects whether the model can be applied to high-speed real-time tasks.

We also performed a comparison using the KITTI dataset. The results are shown in Table 3.11, which shows that the results of STAFNet were on par with those of OIFlow [91], but slightly lower than UPFlow [99] in both the training and test sets. We speculate that the performance gap between Sintel and KITTI is due to the difference in the distribution of their flow statistics. For the DAVIS and Sintel datasets, camera views are slow-moving or stationary in most cases. In contrast, KITTI has high-speed ego-motion issues in a vehicle-moving environment, resulting in

Table 3.11: Comparison on KITTI 2012 Benchmark. We used the average EPE error (the lower, the better) as an evaluation metric for the Sintel datasets. Missing entries ‘-’ indicate that the result is not reported in the compared paper.

Method	UFlow [77]	ARFlow [87]	OIFlow [91]	UPFlow [99]	Ours
Train	1.68	1.44	1.33	1.27	1.33
Test	1.9	1.8	1.6	1.4	1.6
Test-Noc	0.9	-	1.0	-	1.0

a large range of motion fields out of the image boundary, potentially damaging the performance of the global correspondence matching module. Exploring non-linear matching methods could be a promising approach to generalize the global correspondence matching model for datasets like KITTI, suggesting potential avenues for future research.

3.6.4 Augmentation on Natural Image

Benefiting from the unsupervised learning method that does not require labels, we exposed the model to additional training on the DAVIS natural dataset; we randomly extracted images from Sintel and DAVIS as samples at a rate of 50 %, while the rest of the settings were kept the same as in subsection 3.6.1. The extended training not only produced clear motion fields on the DAVIS dataset, but also its generalization performance on the Sintel test set improved, The DAVIS dataset results are qualitatively shown in Figure 3.14, and its generalization performance on the Sintel dataset is shown in the last row of Table 3.10.

3.6.5 Comparison with Prototypes

We evaluate the performance of our models on the Sintel dataset using a supervised learning approach and compare it to the prototype model, GMFlow [187]. To ensure fairness in the comparison, we maintain consistency in the training strategy for both models. Both models are trained directly on the Sintel training set without pre-training and utilize the same weighted sequential loss function as $1e^{-4}$. However, there are structural differences in the model architecture, which results in our training and inference being conducted in a multi-frame environment. For the comparison, we selected GMFlow with only a single inference at a 1/8 scale, named GMFlow-8, which has a similar computational volume to our model. The results are presented in the Table 3.12.

3.7 Conclusion

This chapter explores motion estimation using unsupervised and self-supervised learning methods. We propose a multi-frame-based unsupervised method that enables the model to learn

Table 3.12: Comparison with prototypes in supervised learning; We used the average EPE error (the lower, the better) as an evaluation metric for the Sintel datasets. RT: running time; Param.: memory overhead.

Method	Clean	Final	RT (ms)	Param. (MB)
GMFlow-8 [187]	1.07	1.69	25ms	4.70M
STAFNet	0.90	1.33	28ms	7.04M

motion estimation directly from unlabeled real videos, serving as an alternative training method for our human-aligned model discussed in Chapter 5. Additionally, we incorporate techniques from computer vision and propose several strategies to address challenges in current warping-reconstruction-based methods. Recognizing that motion is a smooth and dynamic process spanning multiple frames, we introduce solutions based on temporal dynamic modeling and present two proposals to enhance the training protocol and model architecture.

Proposal I applies unsupervised learning to long-term dynamic environments using multi-frame scene augmentation strategies. To tackle challenges such as occlusion, brightness variation, and motion blur, we introduce three temporal dynamic training enhancers—DOE, CVE, and SVE—which collectively improve performance by about 20% without adding computational or memory overhead. Combining these techniques, the proposed lightweight model achieves comparable performance to current state-of-the-art results, demonstrating the efficiency of temporal dynamic modeling for unsupervised optical flow estimation.

Building on Proposal I, Proposal II designs a more advanced temporal recurrent structure to complement the earlier approach. Motion estimation is decoupled into apparent matching regions and LMC regions, with the Spatiotemporal Adaptive Fusion Network (STAFNet) introduced for self-supervised motion estimation. Techniques such as dynamic occlusion pseudo-labeling, efficient training algorithms, and confidence description are proposed. Integrating all components, STAFNet achieves state-of-the-art performance on both the Sintel Clean and Sintel Final datasets.

We will show in Chapter 5 that the proposed unsupervised method allows our human-aligned model to learn motion perception with performance comparable to that of supervised methods.

Chapter 4

Limitations of Current Machine Vision Models to Predict Human Motion Perception

Chapter 3 investigates unsupervised learning of motion estimation, incorporating several strategies from the computer vision field. As a key area for studying image-computable models, current methods in the computer vision field, powered by deep learning and deep neural networks, demonstrate excellent performance in existing benchmarks and show promise in real-world applications and industrial fields. Similarly, in visual motion estimation, current computer vision models can accurately estimate object motion and cope with complex scenarios in the natural world. These models, at least at a performance level, provide promising insights for achieving human-aligned capabilities in motion perception. In this chapter, we investigate the (dis)similarities between several representative machine models and human motion perception, testing whether these state-of-the-art models can replicate human visual motion perception.¹

Specifically, we incorporate the *in silico*-psychophysical method to examine how well representative motion estimation models from the computer vision field can explain known psychophysical phenomena of human motion perception. We also analyze the factors accounting for the similarities or dissimilarities between human perception and machine models. Overall, we will show that there is a gap between human perception and model response in some known psychophysical stimuli, which can be explained by structural or inner mechanism misalignment and the generalization problem of the training dataset. The specific findings are listed as follows:

1. We found that most DNN-based models did not generalize well to simple experimental stimuli unless they underwent special hybrid training, which included simple stimuli and non-texture motion patterns in the training set.

¹The supplementary video demo of this Chapter can be found in https://github.com/KUCognitiveInformaticsLab/VideoDemo_Zitang/tree/main/chapter%204

2. DNN-based models cannot account for perceptual motion reversals perceived by humans with contrast-reversed patterns and a missing fundamental grating. However, some of these models can explain the barber-pole illusion and spatial motion integration. In contrast, a motion-energy-based model shows the opposite pattern of results.
3. The interpretation of motion stimuli by DNN-based models is influenced by the training set, training method, and model architecture. Supervised training enables the model to have stronger capabilities than unsupervised learning. Advanced structures, such as Raft, are capable of propagating and integrating motion through recurrent iterations.

4.1 Introduction

The study of motion is a historical and classic topic in human vision. Earlier models, such as the Reichardt model [57], the motion energy model [2], and the gradient-based model [60, 97], explain the mechanisms of human motion perception from different perspectives. With the rise of deep learning, motion perception based on deep neural networks has also achieved significant success in the field of computer vision. In this context, motion perception is defined as optical flow estimation, which is the projection of the apparent motion of objects in a scene onto the image plane of a biological vision system or a visual sensor.

In 2015, FlowNet [38] was the first to adopt convolutional neural networks (CNNs) to handle the optical flow estimation problem. Recognizing the powerful capabilities of CNNs in optical flow estimation, considerable research has been conducted in this field, resulting in the development of many accurate and high-performing models. These artificial neural networks often outperform humans in terms of accuracy on specific datasets and have been highly successful in engineering applications. However, we aim to determine whether DNN-based models can better represent the human visual system and whether they operate similarly to human vision systems using an *in silico* psychophysical approach.

To this end, we tested several representative computer vision models, including:

- FFV1MT [155], a bio-inspired model based on motion energy constraints.
- FlowNet [38], the first DNN optical flow estimation model using encoder-decoder architecture.
- FlowNet2 [68], an engineering-oriented large network composed of several sub-FlowNet.
- RAFT [171], one of the most advanced optical flow estimation architectures,
- ARFlow [87], an unsupervised learning-based optical flow estimation model.
- IRR-PWC [66], a representative model that integrates classical ideas based on coarse-to-fine refinement.

Table 4.1: Summary of different motion estimation models. M-S: multi-scale methods; R-D: recurrent decoding methods; SV: supervised learning; USV: unsupervised learning; Biocomputing methods

Name	Param.	M-S	R-D	SV	USV	Bio
FFV1MT	N.A.	✓	✓		✓	✓
FlowNet	39.18M	✓	✓	✓		
FlowNet2	162.52M	✓	✓	✓		
RAFT	5.26M		✓	✓		
IRR-PWC	6.36M	✓	✓	✓		
ARflow	2.24M	✓	✓		✓	

All models are summarized in Table I. Various stimuli were used to test the responses of these different networks.

4.2 Test on Simple Stimuli

We first adopted the pre-trained models from the original papers. All models were fine-tuned on the Sintel [18] dataset to achieve the highest accuracy reported in their paper.

We generated a drifting Gabor and a simple moving box as test stimuli. Without specific explanation, each stimulus was set to 512×512 with a constant moving speed of 4 pixels per frame by default. For the sinewave, we set a reasonable spatial frequency (32 pixels per cycle). To our surprise, although all models performed well on the Sintel dataset, they had poor generalization performance on simple stimuli commonly used in vision research. Figure 4.1 shows a straightforward scenario where a white box moves rightward against black backgrounds. The color-coding map illustrates the relationship between flow vectors and colorful visualization. All tested DNN-based models predict unstable flows that deviate significantly from human perception. As one of the state-of-the-art optical flow estimation models, RAFT achieved satisfactory performance on the challenging Sintel dataset but could not cope with the simpler stimulus mode here. Besides, different DNN models exhibited different behaviors for the same stimulus despite training on the same dataset. As for the biological-orientated model, despite the low spatial separation ability FFV1MT based on motion energy computing demonstrated a reasonable response of moving direction.

4.3 Mixed Training Strategy

We found that these DNN models all prefer textured regions with rich gradient cues as backgrounds, especially natural textures. To strictly control for confounding elements, the experimental stimuli used in vision studies often consist of unnaturally simple elements. The current

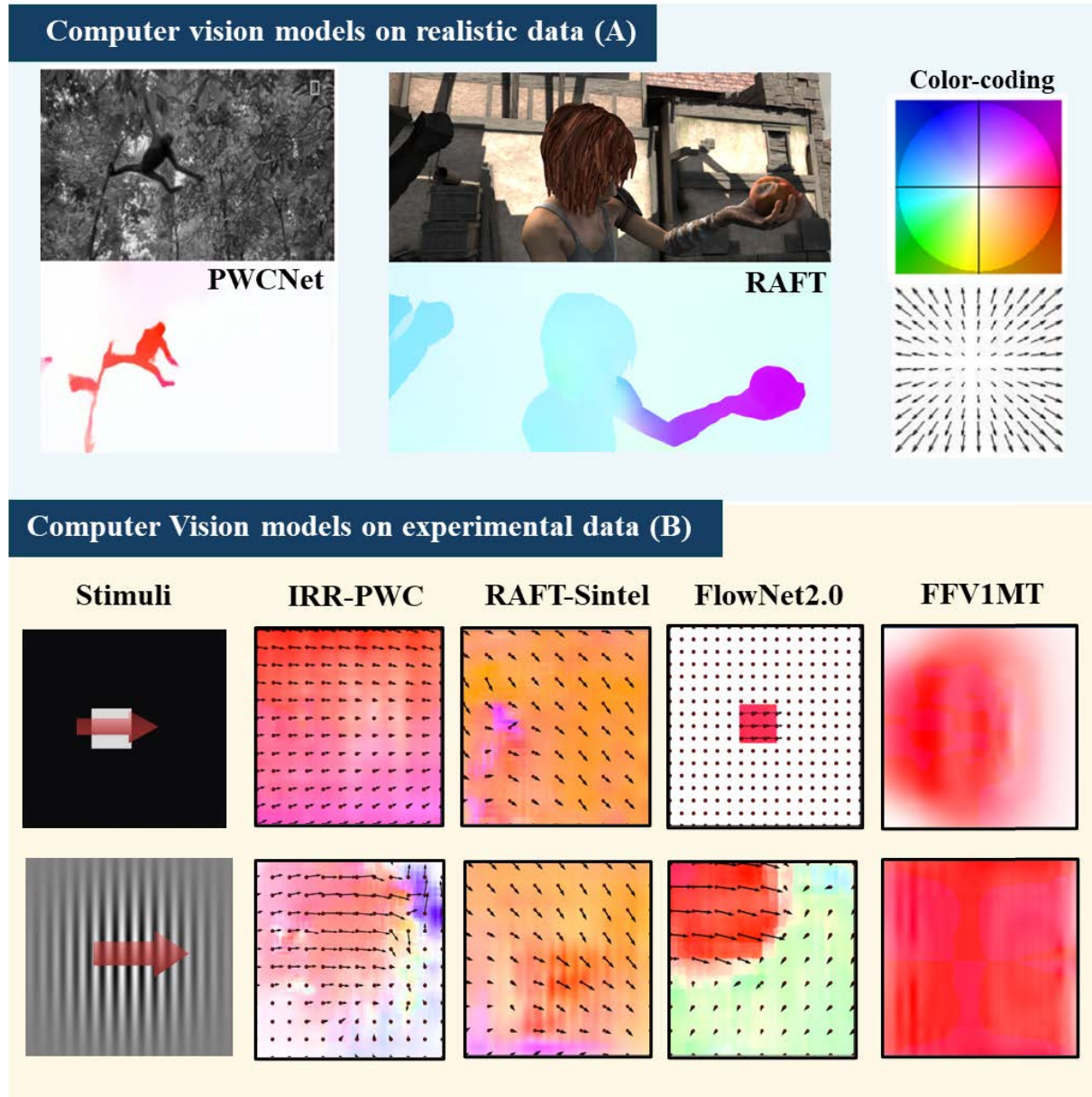


Figure 4.1: (A). Response on the naturalistic scene; (B). Responses on the moving box and drifting Gabor. There is a significant performance gap between the two datasets, and the response of CV models failed to replicate the human perception of these stimuli that are commonly used in vision science

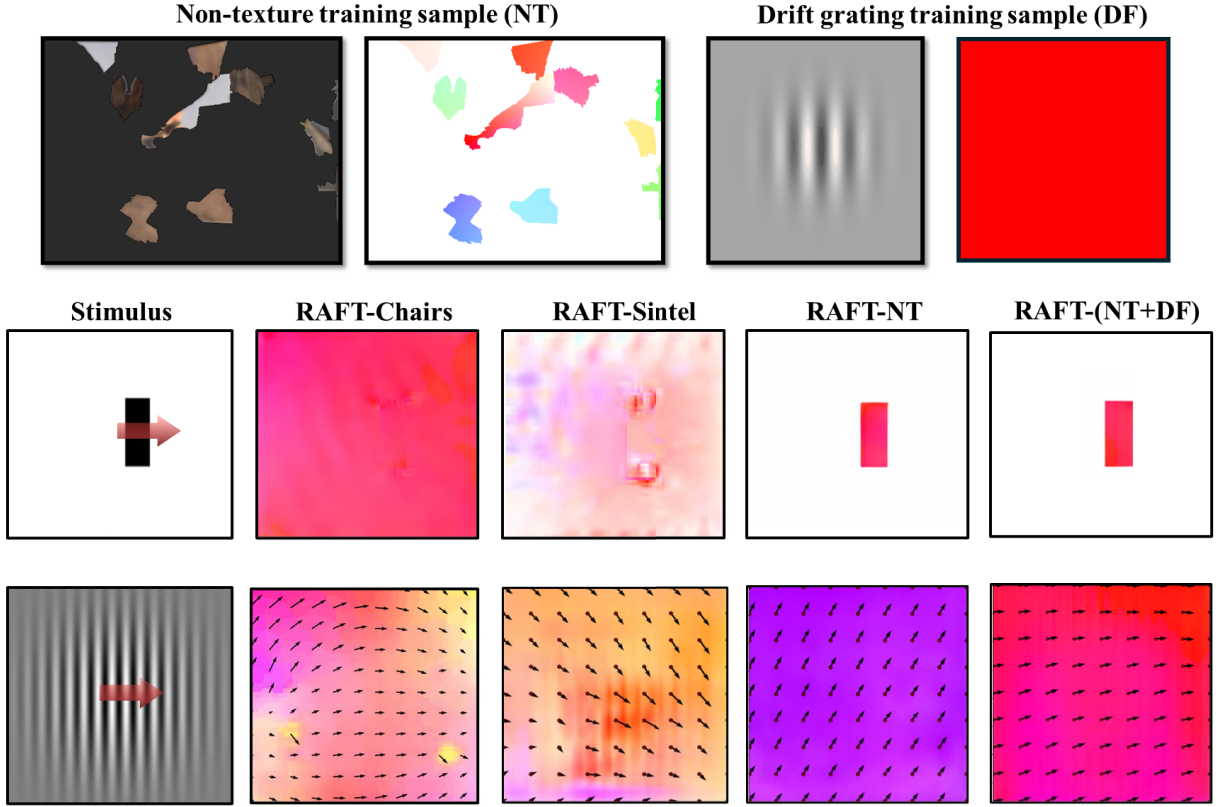


Figure 4.2: Responses of RAFT trained on different datasets. NT: simple non-texture dataset; DF: drifting grating dataset.

motion datasets in the computer vision field do not cover such stimuli, which may result in poor generalization of the models. Almost all existing deep learning models are unable to generalize to simple stimuli used in psychophysics, making it challenging to analyze DNN models using some classical psychophysical methods.

In addition, we trained the RAFT model on a separate dataset. The results suggest that current motion datasets are not comprehensive enough to generalize to the motion stimuli used in vision science. Different dataset distributions also affect the behavior of the network, as illustrated in Figure 4.2. To address this issue and mitigate individual biases associated with different training histories, we developed a mixed training approach. We trained models using multiple datasets in a mixed manner.

Specifically, we combined the MPI-Sintel, KITTI, FlyingChairs, 3D FlyingThings, and two additional designed datasets into a single large dataset. Samples were randomly extracted according to a probability distribution during training. Including non-textured stimuli and drifting gratings enabled the models to generalize to different non-textured conditions while providing a potential "slow-world" Bayesian prior to solving ambiguous one-dimensional motion [184]. As shown in the bottom of Figure 4.2, our hybrid training strategy allowed the model to generalize to both simple non-textured stimuli and complex natural environments. Furthermore, robust

responses to simple stimuli allowed us to analyze the network using standard psychophysical stimuli. In subsequent sections, we use the re-trained model for analysis by default. These re-trained models were employed in the psychophysical studies discussed in the following sections.

4.4 Motion Illusions

Missing Fundamental Illusion:

The human visual system exhibits a strong bias toward detecting motion from the Fourier component, which is demonstrated by the “missing fundamental illusion” [2, 17] in Fig.4.3 (A). This illusion occurs when the first spatial harmonic of a square wave is removed, resulting in a reversal in the perceived motion direction. FFV1MT shows a similar interpretation to humans because it is designed to detect motion from Fourier components. In contrast, CV models, which are designed to infer optical flow based on structural correspondence, do not exhibit this bias. As shown in Fig.4.3 (A), RAFT’s response is biased to structural motion, which occurs to humans only under specific conditions, where a high-level attentive tracking system dominates. [48].

Reversed Phi:

In Fig.4.3 (B), humans perceive a continuous motion for a four-stroke reversed phi in which image shift direction and luminance contrast are reversed at even numbers of frame transition [5]. The DNN-based models (e.g., RAFT) can not perceive continuous motion for a four-stroke stimulus and generate an opposite rotating direction for frame transitions with reversed phi (see [190] for a similar observation). The FFV1MT, on the other hand, can generate continuous motion similar to humans’ response, which indicates that the energy model is easier to generate continuous motion than CV models.

Second-order Motion:

There are many kinds of second-order motion that could be perceived by humans, where we choose drift-balanced motions [25] and theta motion [197] as stimuli. The experimental results show that all models are unable to produce significant responses to second-order motions. The response of supervised DNNs is basically close to zero. The unsupervised model, ARFlow, and FFV1MT can detect some noisy motions that differ significantly from humans’ perception (Fig.4.3, C).

Barber pole illusion:

The barber pole illusion is a well-known example of how the spatiotemporal energy-based optic

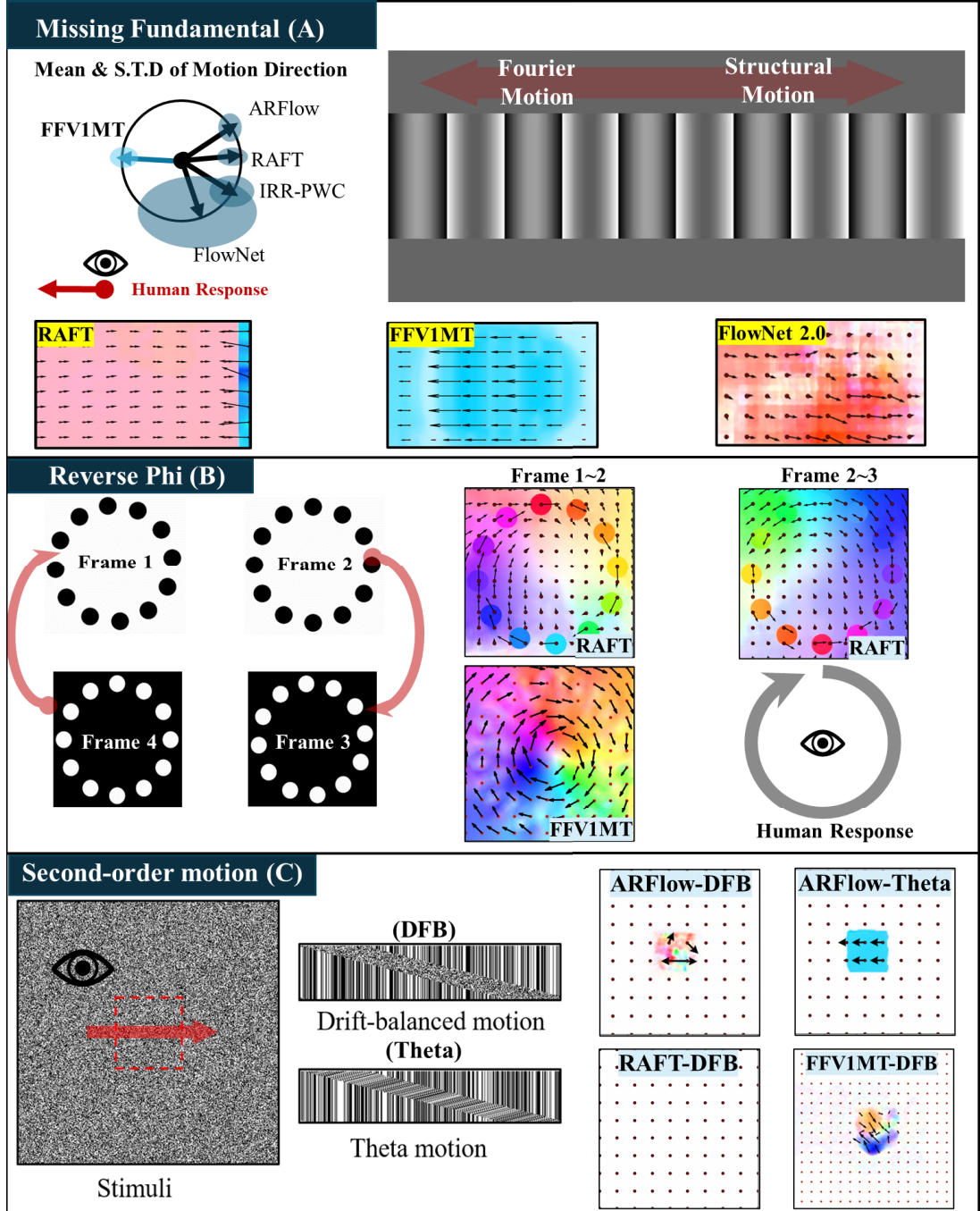


Figure 4.3: Responses on classical motion stimuli & illusion. (A) Missing fundamental illusion: humans demonstrate a strong illusion of a leftward motion, while the majority of DNN models detect the rightward motion; (B) Reverse Phi illusion: humans perceive a continuous clockwise motion for a four-stroke reverse phi, while RAFT detected a reversed direction between frames 1-2 and 2-3; (C) We choose drift-balanced motions [25] and theta motion [197] as second-order motion stimuli.

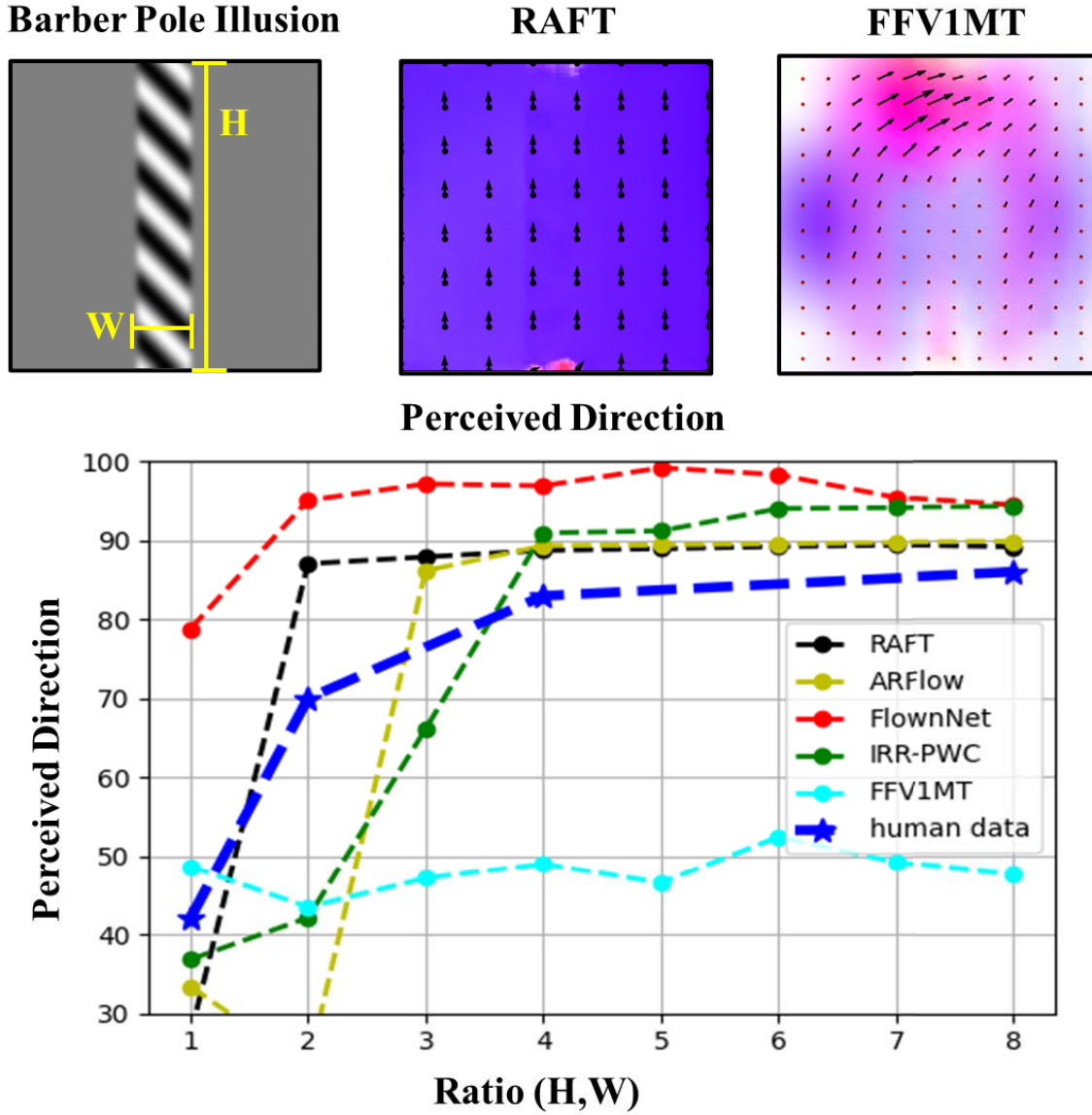


Figure 4.4: Barber pole illusion. We show the result of RAFT (Computer Vision), FFV1MT (Motion energy-based model), and Human response with the H:W-ratio growth. Generally, computer vision models can perceive the high-order feature motion, but the motion energy model only captures the Fourier motion.

flow undergoes higher-level distortions to facilitate tracking of feature motion [166]. Specifically, as the height-width ratio of visual stimuli varies, humans' perceived direction shifts from oblique to vertical [45], as illustrated in Fig.4.4. Generally, DNN models demonstrate a similar shift in perceived direction, suggesting that the model can track features beyond simple frequency-based motion. In contrast, FFV1MT only captures the low-level Fourier motion.

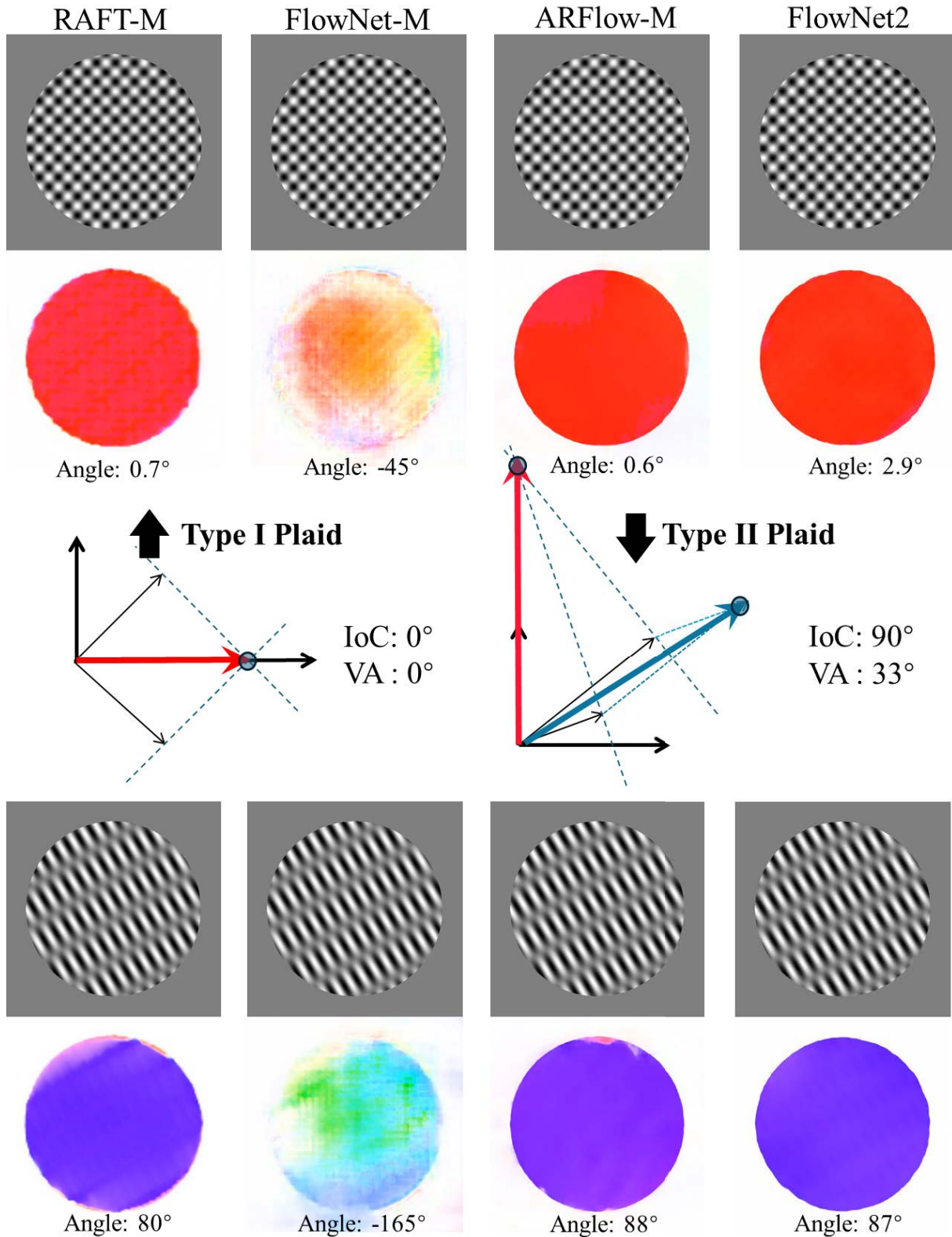


Figure 4.5: Evaluation of motion integration using plaid stimuli; {}-M represents the models that are retrained in a mixed manner.

4.5 Motion Integration

4.5.1 Motion Integration on Plaid

Early visual motion processing must resolve the ambiguity of local motion signals, which are detected by one-dimensional motion sensors with spatially oriented receptive fields [44]. Inferring two-dimensional vectors from local one-dimensional motion sensor outputs is an essential function of the visual motion perception system. Component stimuli (sinusoidal gratings with a single frequency component and moving direction) and plaid stimuli (consisting of two components with different parameters) have proven effective for examining this integration perceptually. Psychophysical studies have demonstrated that humans can integrate the motions of plaid components under various conditions. Theoretical approaches for combining these vectors include the intersection of constraints (IoC), which is accurate but complex to implement physiologically [116], and a weighted vector-average (VA), which effectively approximates correct motion in most situations. The latter method's predictions often align with errors observed in visual processing during brief presentations or when the image is degraded by low contrast or noise [160]. Figure 4.5 investigated motion integration using Type I and Type II plaids, respectively. Basically, the majority of deep neural network models demonstrated the ability to integrate the two components and generate a stably integrated direction, except for FlowNet, which could not generalize this type of motion stimuli.

For Type I plaids, the models replicated motion directions close to the predictions of both the IoC and VA. For Type II plaids, the models exhibited directions primarily oriented towards the IoC prediction, but with minor deviations towards the VA prediction across different models. This pattern also captured a similar tendency observed in human perception.

4.5.2 Motion Integration with Frame and Occlusion Constraints

More complicated scenarios are designed for testing motion integration with frame constraints and occlusion/ layer cues. Our investigation reveals the existence of a motion integration mechanism in some DNN models, as illustrated in Fig. 4.6. First, as exemplified by the RAFT and ARFlow, the supervised learning models exhibit stronger motion integration capabilities than unsupervised learning models. In addition, the RAFT shows stronger spatial motion pooling for local drifting Gabor (ambiguous motion) than for local drifting plaids (unambiguous motion). This indicates the model's tendency to prioritize local integration of motion signals over global integration. Once the local ambiguity is resolved, the global integration process is suppressed, the mechanism behind which is similar to the adaptive integrating motion strategy observed in humans [4]. In the third column of the figure, where a rotating box is presented, the RAFT model exhibits good integration of the four edges of the frame, thereby resolving the aperture problem. However, when occlusion occurs in the fourth column, all models lose their spatial integration ability, deviating from human perception [108].

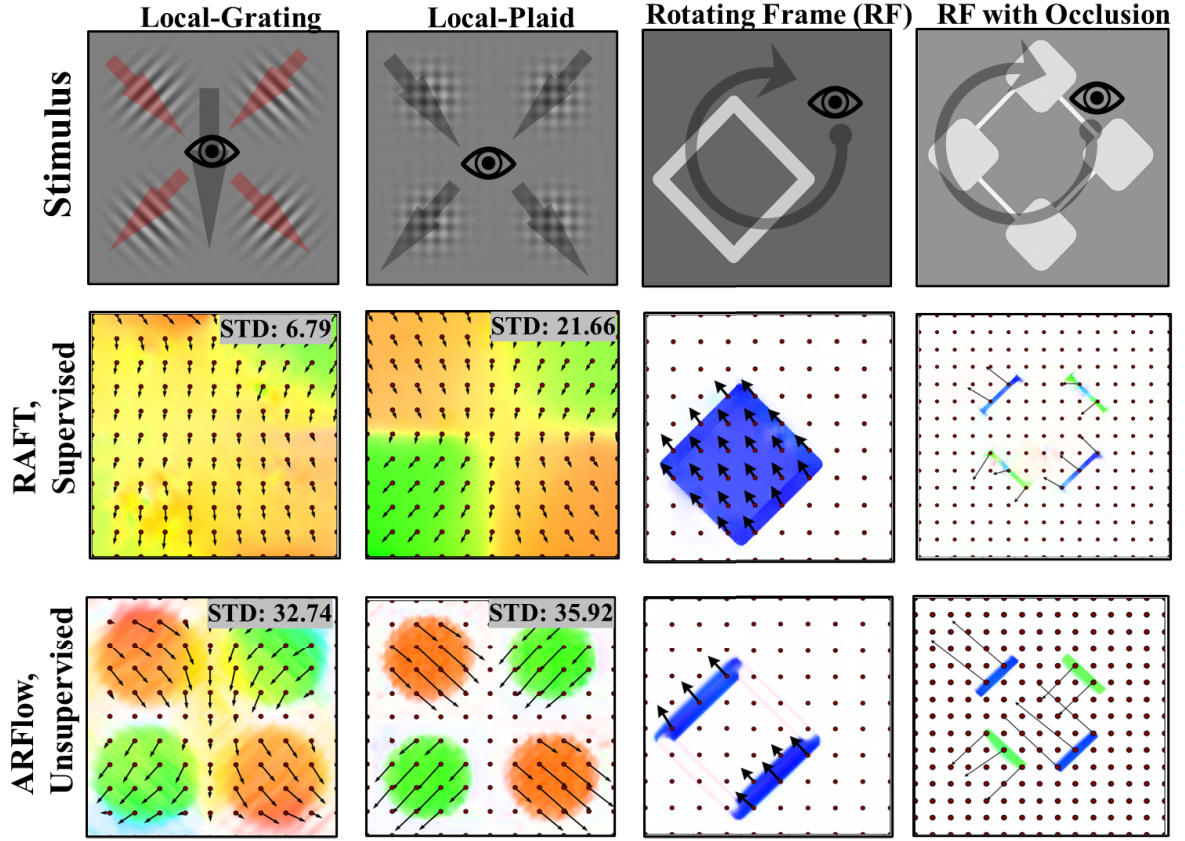


Figure 4.6: Test on Global Motion integration with frame constraints and occlusion cues. Gray arrows indicate the direction perceived by humans. Red arrows indicate the direction of physical motion.

4.5.3 Motion Integration with Recurrent Process

Figure 4.7 (A) illustrates the performance of DNN models when tasked with interpreting the motion of a translating hollow diamond. Notably, FlowNet2.0 exhibits a robust capability to synthesize local motion from the diamond’s four segments, suggesting that deeper architectural frameworks may augment a DNN’s capacity for motion integration. In contrast, the RAFT model’s response deviates from that of the other models. It integrates motion not only from the four segments but also extends this integration into the diamond’s inner regions, potentially indicating advanced motion integration mechanisms exclusive to RAFT.

To explore this phenomenon, we further analyzed the recurrent decoding process in the RAFT model. Our analysis reveals that its distinctive capabilities likely stem from the recurrent decoding strategy, which leverages a Gated Recurrent Unit (GRU) to iteratively refine optical flow by looking up local correspondences. The second row of Figure 4.7 visualizes this dynamic, showing how motion initially recognized at the boundaries progressively permeates the inner regions through subsequent decoding iterations.

During the final layer of the encoding phase, visualization of the average activation within

the feature map reveals that the model responds solely to the edges of the diamond, with no activation in the inner areas. However, as the decoding phase advances, the recurrent process facilitates a gradual transfer of motion energy from the peripheral edges to the core areas enclosed by the frame. Although this propagation pattern diverges from typical human responses, it underscores the inherent ability of models equipped with recurrent decoding blocks to convey motion information across extended spatial distances.

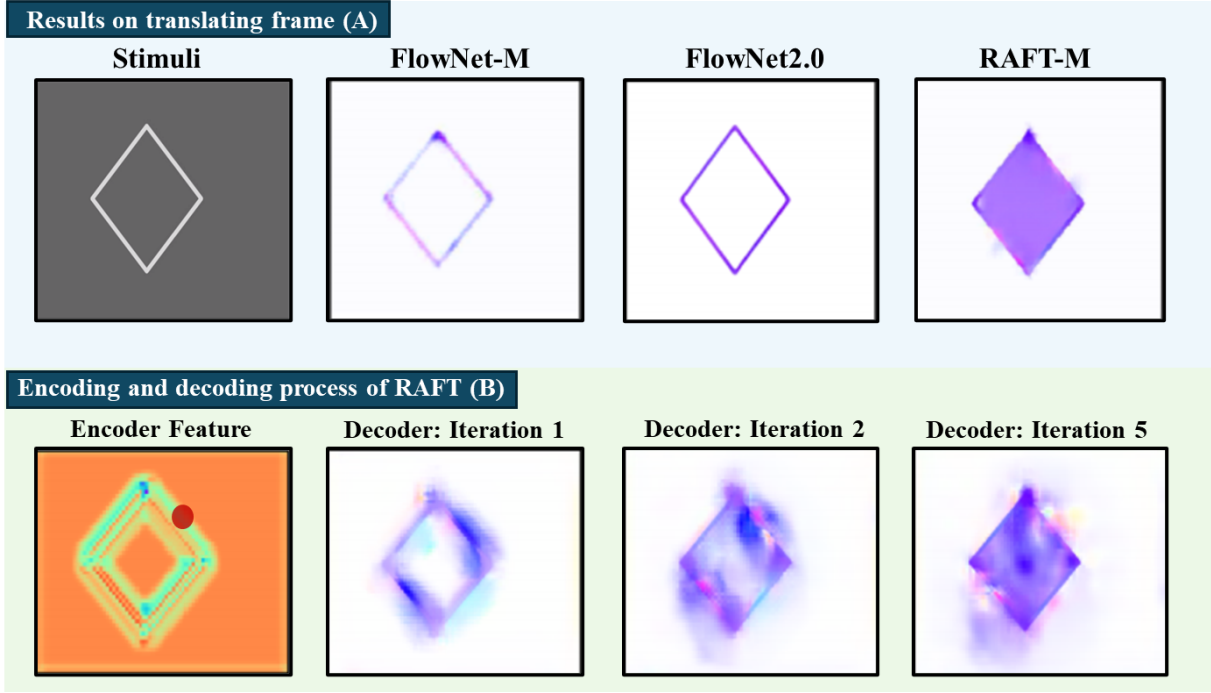


Figure 4.7: Motion integration with recurrent process. (A) response of three DNN models on a translating Hollow diamond; (B) the recurrent decoding process of RAFT; {}-M indicates models that retrain in a mixed manner.

4.6 Conclusion and Discussion

In this chapter, we verified and summarized the performance of current DNN-based motion models compared to human visual motion perception. Through multiple psychophysical stimuli, we identified similarities and dissimilarities between computer vision models and human perception of motion. Our findings provide insights on how to effectively employ deep learning technologies to model the human motion perception system.

Firstly, we found that the training dataset and strategy significantly affect the behavior of structurally identical models. Similarly, models with different architectures exhibit different behaviors even when trained with the same strategy. Secondly, many deep learning models prefer naturally distributed textures and show poor generalization toward simple psychophysical stimuli with non-textured backgrounds. To address this, we propose a mixed training strategy

coupled with two self-designed datasets to enhance the generalization of DNNs on simple stimuli. Thirdly, we found that models with recurrent decoding blocks, such as RAFT, possess a specific mechanism for motion integration, primarily occurring during the decoding phase. Finally, none of the models exhibited the same perception of second-order motion as humans. Supervised learning-based models showed almost no response, while unsupervised learning-based models responded to local pattern changes, differing from human perceptions.

Additionally, we found that most existing models are based on a two-frame correspondence matching architecture, focusing on improving optical flow accuracy rather than predicting human responses. However, such a design misaligns with the mechanisms used in the human vision system, particularly lacking consideration of the temporal dynamics of motion perception systems. As a result, these DNN models are unable to explain certain motion phenomena, such as the missing fundamental illusion, motion aftereffect (MAE), and four-stroke reverse phi.

Chapter 5

Human-aligned Visual Motion Processing Model: First-order motion

Visual motion processing is essential for animals and AI agents to perceive and interact with dynamic environments. With the final goal of building a human-aligned motion process, the previous Chapter 4 investigated the similarities and dissimilarities between human motion perception and existing computer vision models. Although computer vision models demonstrate powerful results in accurately inferring physical motion at the sub-pixel level, they exhibit significant misalignment in structural consistency and plausibility with biological systems. Additionally, their responses to multiple experimental stimuli deviate substantially from human perception, indicating a considerable gap between the two approaches.

In another branch of human vision science, extensive cognitive neuroscience and psychophysical studies attempt to explain and model the inner mechanisms of human visual processing. However, the majority of these studies focus on concise cases to understand parts of the mechanism, using simple stimuli like drifting gratings, plaids, and random dots. Few studies focus on image-computable models that can extract informative motion flow from complex natural scenes. Models with such capabilities in a manner consistent with human visual processing have yet to be established.

We propose an image-computable model of human motion perception that bridges the gap between biological and computer vision models, combining insights from both fields. We divide motion perception into two parts: first-order and higher-order motion processing. This chapter focuses on first-order motion, while the higher-order (second-order) motion problem will be discussed in Chapter 6.

Specifically, in this chapter, we introduce a novel two-stage approach that combines trainable motion energy sensing with a recurrent self-attention network for adaptive motion integration and segregation. This model architecture aims to capture the computations in V1 and MT, the

core structures for motion perception in the biological visual system, while providing the ability to derive informative motion flow for a wide range of stimuli, including complex natural scenes. In silico neurophysiology reveals that our model’s unit responses are similar to mammalian neural recordings regarding motion pooling and speed tuning. The proposed model can also replicate human responses to a range of stimuli examined in past psychophysical studies. Experimental results on the Sintel benchmark demonstrate that our model predicts human responses better than the ground truth, whereas state-of-the-art computer vision models show the opposite. Our study provides a computational architecture consistent with human visual motion processing, although the physiological correspondence may not be exact.

5.1 Introduction

Visual motion perception is essential not only for humans and animals to perceive and interact with the world but also for artificial agents to process various dynamic visual tasks. As such, visual motion estimation has been extensively studied by both biological vision and computer vision research communities. The key issue lies in estimating optical flow, an array of instantaneous image motion vectors [49, 176].

Vision science has revealed that optical flow estimation in the human visual system (HVS) is primarily served by a pathway that includes the primary visual cortex (V1 [63, 64]) and middle temporal (MT [22, 146]) area. A significant portion of neurons in area V1 are sensitive to the direction of local motion, while those in area MT integrate and segregate local motion signals for global flow interpretation. The MT process also helps to overcome the aperture problem [130]. Several computational models of area MT have been proposed [152, 184], but mechanisms that can fully encapsulate a variety of motion integration abilities of MT neurons and/or human perception have yet to be developed [129]. Furthermore, previous studies tested the models using only simple artificial stimuli. It remains challenging to create a human-like optical flow extraction mechanism that can derive informative motion flow for a wide range of stimuli including complex natural videos.

On the other hand, optical flow estimation has recently made remarkable progress in the field of computer vision. FlowNet [38] pioneered the use of fully convolutional neural networks for dense optical flow estimation, and various approaches based on DNNs have emerged subsequently [68, 165, 171, 187]. Owing to the powerful representation ability of DNNs, these models outperform humans in estimating the ground-truth optical flow of natural scenes [189]. Consequently, one might expect DNN-based optical flow estimation algorithms to become promising models of human visual motion processing, similar to how ImageNet-trained DNN models provide good computational models of human object recognition [188]. However, since computer vision optical flow models are designed solely to find local image correspondences between pairs of frames on the image coordinates [38], they cannot explain systematic or adaptive deviations of human perceived motion from local ground truth (GT) [189]. Moreover, existing DNN mod-

els often exhibit instability when dealing with non-textured stimuli commonly used in vision science [167].

In this chapter, we leveraged the flexibility of DNNs to construct an image-computational model of human motion processing. While recent studies successfully used DNNs to elaborate the understanding of the neural mechanism of visual motion processing [115, 118, 143, 144, 161], we aimed to make a model that can explain a broader range of physiological and psychophysical phenomena, including those whose neural mechanisms are not yet clear. From an engineering standpoint, we aimed to make a human-aligned optic flow algorithm while maintaining a flow estimation capability comparable to the state-of-the-art (SOTA) CV models. Our model extracts dense informative motion flows for a wide range of inputs in a way consistent with physiologically measured neural responses and psychophysically measured human motion perception. It consisted of two stages. The first stage, which mimicked the function of V1, comprised of neurons with multi-scale spatiotemporal filters to extract local motion energy. Unlike previous models of V1 [2, 134, 152], the filter tunings were learnable to fit natural optic flow computation. The second stage, which mimicked the function of MT, recurrently integrated local motion signals and solved the aperture problem. We constructed an undirected fully connected graph from the map of local motion energy and used the attention mechanism [37, 178] for adaptive global motion integration and segregation.

We evaluated the performance of our model from several aspects. In in-silico neurophysiology, we found that our model’s neurons exhibited direction and speed tunings similar to those observed in mammalian physiological recordings in V1 and MT [117, 135]. In simulations of psychophysical findings, our model showed good generalization from simple artificial stimuli (e.g., drifting Gabor) to complex naturalistic scenes. Our model produced human-like responses for several conventional motion stimuli and illusions, including global motion pooling and the barber-pole illusion. Furthermore, the model’s response to natural scenes was closer to that of humans in comparison to other computer vision models. Our two-stage model provides a computational architecture consistent with human visual motion processing, although the physiological correspondence may not be exact. This achievement is not only valuable in terms of neuroscientific understanding of human visual computation but also for the development of human-aligned machine vision that stably recognizes the world as humans do.

5.2 Molding of Two-stages Motion Perception System

5.2.1 First-stage: Local Motion Energy Computation

Spatiotemporal separable Gabor filter: Since we aimed at an image-computable model, the input is a sequence of grayscale images $\mathbf{S}(\mathbf{p}, \mathbf{t})$ for all spatial positions $\mathbf{p} = (x, y)$ within the image domain Ω and for all times $\mathbf{t} > \mathbf{0}$. The goal of the first-stage neuron is to capture local motion energy at a specific spatiotemporal frequency, which is associated with the function

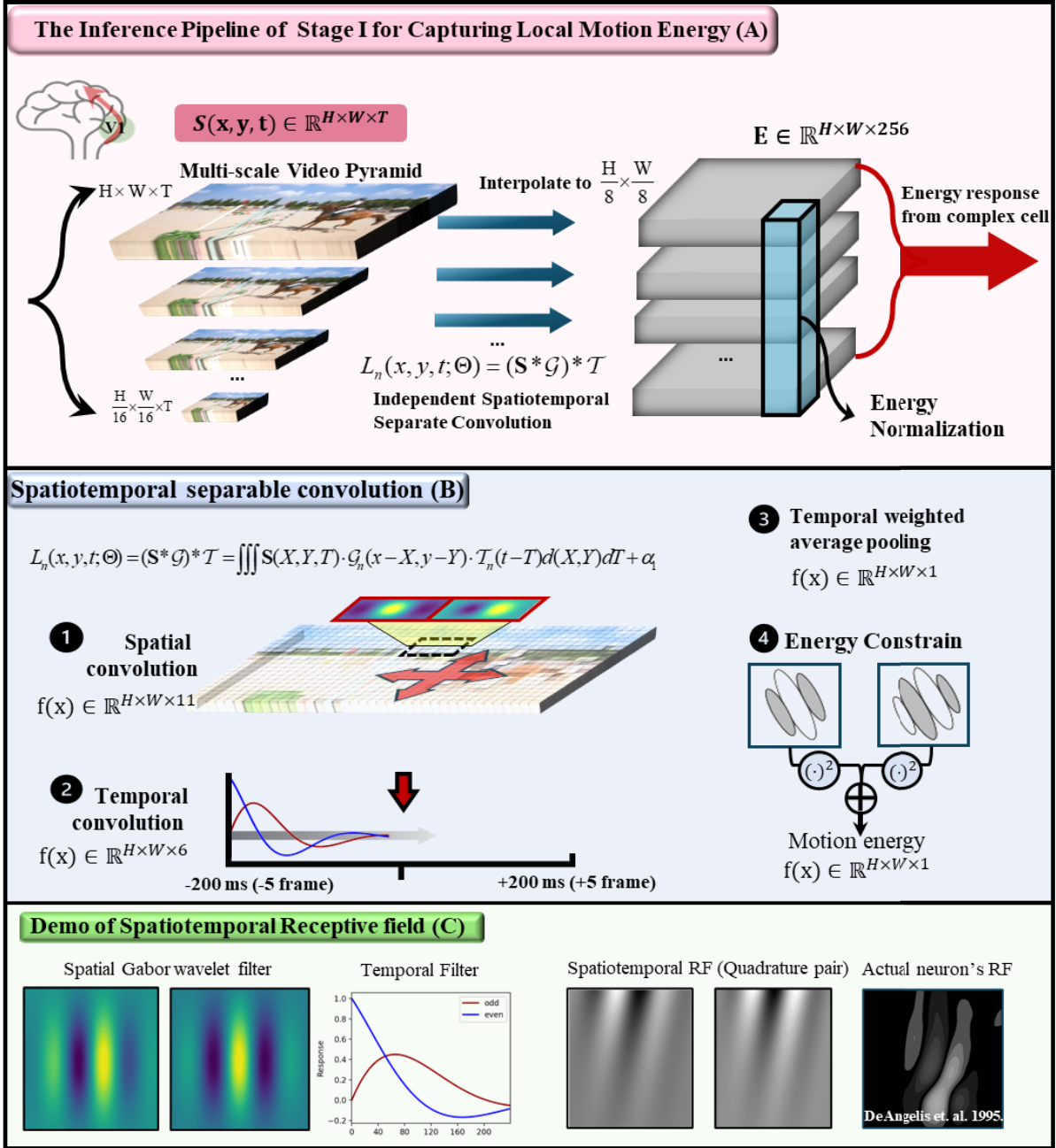


Figure 5.1: Molding of motion perception system. **(A)**: The first stage is built using a group of trainable motion energy units to capture local motion energy; **(B)**: Motion energy calculation based on spatiotemporal separable filters, a sub-block of **(A)**; **(C)**: Demo of spatiotemporal separable filters, a sub-block of **(B)**.

of a direction-selective neuron in the V1 cortex. The responses of neurons can be modeled as 3D Gabor filters [75, 76]. Gabor filters are known to be optimal in the sense that they achieve maximal resolution in both the spatiotemporal and associated frequency domains [16]. To save computational complexity, we decomposed 3D spatiotemporal filters into filters separable in space and time. The spatial component is described by 2D Gabor filters $\mathcal{G}(\cdot)$, and the temporal

component $\mathcal{T}(\cdot)$ is described by a 1D sinusoidal function with exponential decay. Specifically, given $x' = x \cos \theta + y \sin \theta$ and $y' = -x \sin \theta + y \cos \theta$ within the receptive field, the impulse responses of spatial and temporal complex filters are defined as:

$$\begin{cases} \mathcal{G}(x, y; \textcolor{red}{f}_s, \theta, \sigma, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right) \exp(i(2\pi f_s x')), s.t. \{x, y \mid (x^2 + y^2 \leq R^2)\} \\ \mathcal{T}(t; \textcolor{red}{f}_t, \tau) = \exp\left(-\frac{t}{\tau}\right) \exp(2\pi i(f_t t)), s.t. \{t \mid 0 \leq t < T\} \end{cases} \quad (5.1)$$

The parameters in red are designed to be trained to fit the dataset, where f_s and f_t denote the spatiotemporal frequency tunings of the filter with the preferred speed $v \propto \frac{f_t}{f_s}$, and θ determines the preferred moving orientation; σ and γ control the shape of the Gabor filter, and τ controls the degree of attenuation of the temporal impulse response. All parameters are subject to certain numerical constraints, such as θ being limited to $[0, 2\pi)$ to avoid redundancy; f_s and f_t are limited to less than 0.25 pixels per frame to avoid spectrum aliasing, etc. The response of a simple direction-selective cell L_n to a video stimuli $\mathbf{S}(\mathbf{p}, \mathbf{t})$ can be computed via separate convolutions:

$$L_n(x, y, t; \Theta) = (\mathbf{S} * \mathcal{G}) * \mathcal{T} = \iiint \mathbf{S}(\mathcal{X}, \mathcal{Y}, \mathcal{T}) \cdot \mathcal{G}_n(x - \mathcal{X}, y - \mathcal{Y}) \cdot \mathcal{T}_n(t - \mathcal{T}) d\mathcal{X} d\mathcal{Y} d\mathcal{T} + \alpha_1 \quad (5.2)$$

where α_1 can be learned as spontaneous firing rates. Further, local motion energy is captured by a phase-insensitive complex cell in the V1 cortex, which computes the squared summation of the response from a pair of simple V1 cells with approximately orthogonal spatiotemporal receptive fields [2]. We denote the pair of orthogonal (even and odd) simple V1 cells as:

$$\begin{cases} L_n^o(x, y, t; \Theta) = (\mathbf{S} * \text{Im}[\mathcal{G}]) * \text{Re}[\mathcal{T}] + (\mathbf{S} * \text{Im}[\mathcal{G}]) * \text{Im}[\mathcal{T}] \\ L_n^e(x, y, t; \Theta) = (\mathbf{S} * \text{Re}[\mathcal{G}]) * \text{Re}[\mathcal{T}] - (\mathbf{S} * \text{Im}[\mathcal{G}]) * \text{Im}[\mathcal{T}], \end{cases} \quad (5.3)$$

where $\text{Re}(\cdot)$ and $\text{Im}(\cdot)$ extract the real and imaginary parts of a complex number; $*$ denotes convolution operations. Then, the response of a complex cell L_n^c is obtained from a combination of the quadrature pair of the simple cells using the motion energy formulation:

$$L_n^c(x, y, t; \Theta) = (L_n^o(x, y, t; \Theta))^2 + (L_n^e(x, y, t; \Theta))^2 \quad (5.4)$$

Multi-scale Wavelet Processing: The convolution kernel of our spatial filter has a fixed size of 15×15 , which imposes a physical limitation on the receptive field of each unit. To enhance the flexibility of the receptive field size, we employed a multi-scale processing strategy, as shown in Fig. 5.1 (A). Specifically, we constructed an image pyramid consisting of eight images scaled linearly from $H \times W$ to $\frac{H \times W}{16}$. In total, 256 independent complex cells are deployed across different scales, with the lower scale having a larger receptive field and a preference for a faster motion speed. This approach is computationally efficient for capturing large-scale displacements in engineering applications [15, 164], and enables the representation of different groups of cells sensitive to short- and long-distance motions [20]. The N complex cells $\{L_n^c\}_i^N$ capture motion energy on multiple scales. We applied energy normalization to each cell to ensure consistent energy levels:

$$\hat{L}_n^c(t) = \frac{\textcolor{red}{K}_1 L_n^c(t)}{\sum_{i=1}^N L_i^c(t) + \textcolor{red}{\sigma}_1}, \quad (5.5)$$

where σ_1 is the semi-saturation constant of the normalization, and $K_1 > 0$ determines the maximum attainable response. We interpret the response, denoted by $\hat{L}_n(t)$, as the model equivalent of a post-stimulus time histogram (PSTH), which is a measure of the neuron’s firing rate. Physiologically, such responses could be computed via inhibitory feedback mechanisms [19, 58]. Considering the spatial arrangement of images, it is expected that motion energy responses should exist at each spatial location generated by the same complex cell groups. Bilinear interpolation is used to resize the multi-scale motion energy into the same spatial size $= \frac{H \times W}{8}$. In the context of DNNs, this is mainly to balance the trade-off between the spatial resolution and computational overhead, and the final output of the first stage is a 256-channel feature map $\mathbf{E} \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8}}$ that captures the underlying local motion energy, which partially characterizes the cellular patterns of the V1 cortex in a computational manner [2].

5.2.2 Second-stage: Global Motion Integration and Segregation

The receptive field size of first-stage neurons limits their ability to capture only local motion. Advanced spatial integration is an essential requirement for a motion perception system to solve the aperture problem [130]. Spatial integration could involve several mechanisms, such as object recognition and segmentation [50], depth estimation [126], contour structure inference [94], etc., which rely on substantial prior information that may go beyond the capabilities of classical modeling methods. DNNs, with their massive number of parameters and flexibility, offer a suitable solution. However, the spatial integration of local motion requires more flexible connectivity relations than what can be achieved by general convolutions, which are limited by their local receptive field [183]. To overcome this limitation, we propose a computational model based on the attention mechanism and recurrent processing to model the function of motion integration.

Constructing a Graph Structure: First, to establish more flexible connections between cells, we discard the Euclidean space structure within the image and construct topological spaces with an undirected weighted graph $\mathbf{G} = \{\mathbf{V}, \mathbf{A}\}$, where $\mathbf{V}$ is the set of nodes; $\mathbf{A}$ represents the adjacent matrix based on the graph. Each spatial location $p(i, j)$ is treated as a node, and the feature of each node is derived from the whole set of local motion energies $\mathbf{E}(i, j) = \{\hat{L}_n^c(t)\}_{i=1}^{256}$. The connection between any pair of nodes is computed using a specific distance metric, and strong connection relationships are formed between nodes whose local motion energy patterns are similar.

Specifically, given a reshaped feature map $\mathbf{F} \in \mathbb{R}^{HW \times 256}$, the features at each of its locations are first locally embedded into the graph space, as $\varphi(\mathbf{F}) = \text{GELU}(\mathbf{F} \cdot \mathbf{W}_{\Theta 1}) \cdot \mathbf{W}_{\Theta 2}$, where $\mathbf{W}_{\Theta} \in \mathbb{R}^{256 \times 256}$ is a group of trainable parameters. The distance between any pair of nodes (i, j) is calculated by the cosine similarity, which is similar to the self-attention mechanism in current transformer structures [37, 178, 180]. We employ the adjacency matrix $\mathbf{A} \in \mathbb{R}^{HW \times HW}$ to represent the connectivity of the whole topological space, and $\mathbf{A}$ is a symmetric semi-positive

definite matrix defined as:

$$\mathbf{A}(i, j) = \mathbf{A}(j, i) = \frac{\varphi(\mathbf{F})_i \cdot \varphi(\mathbf{F})_j}{\|\varphi(\mathbf{F})_i\| \|\varphi(\mathbf{F})_j\|}. \quad (5.6)$$

We perform exponential scaling of the connections between graphs using the matrix $\mathbf{A}$, given by $\exp(\mathbf{A}s)$, where s is a learnable scalar restricted to the range (0,10) to avoid overflow. The smaller s , the smoother the connections across nodes and vice versa. Finally, a symmetric normalization operation is utilized to balance the energy, given by

$$\mathbf{A} := \mathbf{D}^{-\frac{1}{2}} \exp(s\mathbf{A}) \mathbf{D}^{-\frac{1}{2}},$$

where $\mathbf{D}$ is the degree matrix with

$$\mathbf{D} = \text{diag} \left(\left\{ \sum_j \exp(s\mathbf{A}_{i,j}) \right\}_{i=1}^n \right)$$

. As such, an energy-normalized undirected graph structure is constructed, as illustrated in the top side of Fig. 5.2 (B). Intuitively, this adjacency matrix represents the neuron’s affinity or connectivity within the space, with strong and global connections built across neurons with related motion responses.

Recurrent Integration Processing: Recurrent neural networks (RNNs) are often used to simulate neurons in the brain, as they are flexible in modeling temporal dependencies and feedback loops, which are fundamental aspects of neural processing in the brain [149]. We use a recurrent network, rather than multiple feedforward blocks, to simulate the process of local motion signals being gradually integrated into MT and eventually converging to a stable state.

As shown in Fig. 5.2 (A), the local motion energy from the first stage is divided into two recurrent streamlines. One is the motion energy $\mathbf{E} \in \mathbb{R}^{H \times W \times 256}$, which is continuously updated in the loop, while the other is embedded in the attention space to generate the graph adjacency matrix to control motion integration, denoted as $\mathbf{F} \in \mathbb{R}^{H \times W \times 256}$. In each iteration, the adjacency matrix is first constructed using $\mathbf{F}$. Subsequent motion integration is achieved through a simple matrix multiplication, which is computationally similar to the information propagation mechanism in transformers [178, 180] and can also be considered a simplified version of graph convolution [82]. The integrated motion information is passed through two independent Conv-GRU blocks to update the motion energy $\mathbf{E}$ and feature $\mathbf{F}$, respectively. The Conv-GRU represents a gated recurrent unit [26] implemented in a convolutional manner, and we adopt a spatiotemporal separable approach following RAFT [171]. The motion integration process approximates the ideal final convergence state of the motion energy $\mathbf{E}_k \rightarrow \mathbf{E}^*$ through recurrent iteration.

Decoding the Motion Flow: We adopt the same strategy of decoding the 2D optical flow in each iteration. Initially, the integrated motion is $\mathbf{E}$ projected back to the energy space with a positive value using a square operation, followed by an energy normalization operation:

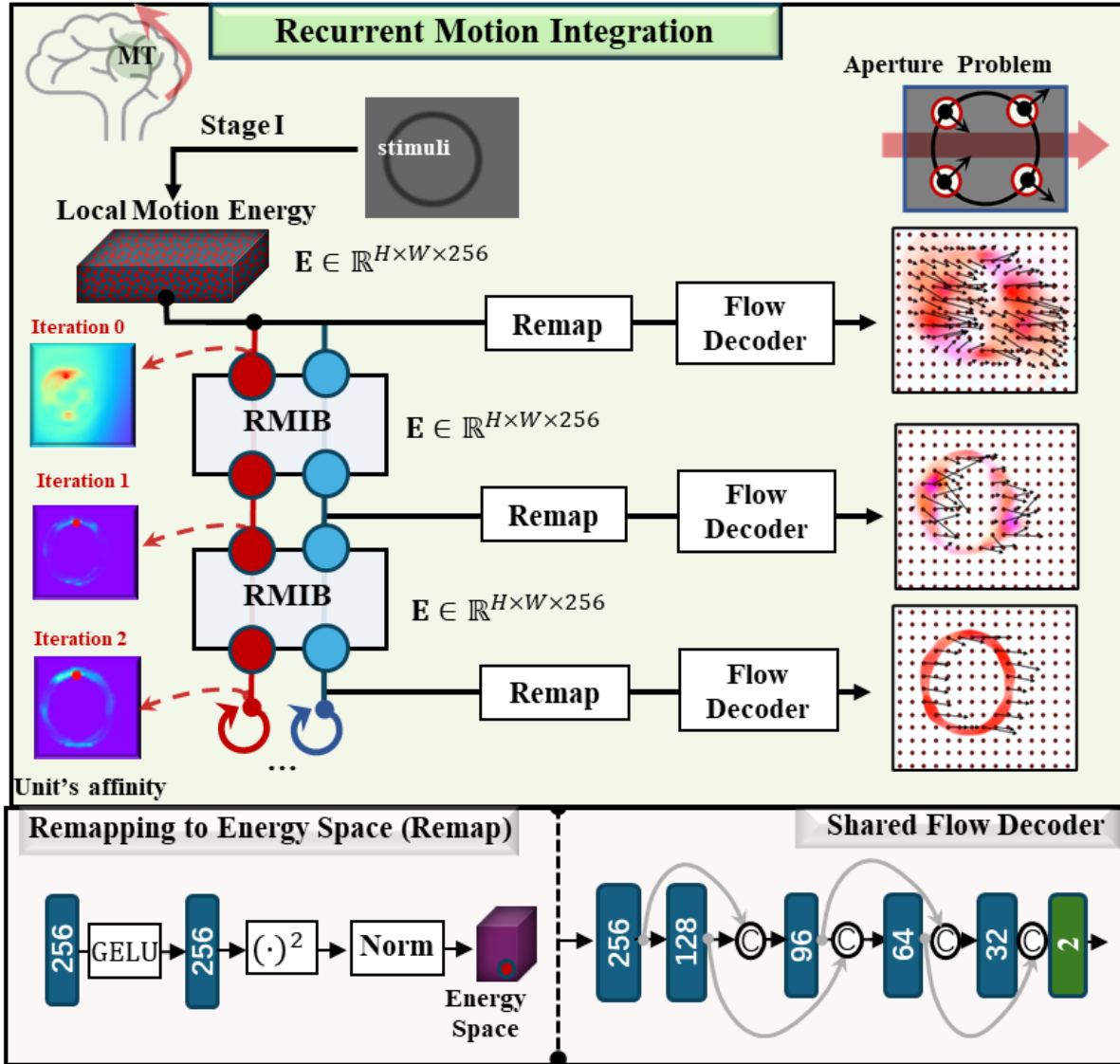


Figure 5.2: Molding of motion perception system (A): The second Stage is constructed by using a self-attention mechanism and recurrent processing to simulate the function of global motion integration. We use an identical flow decoder to visualize the dense optical flow at each iteration. See Figure 5.3 for details of RMIB and legend.

$$\hat{\mathbf{E}}(i, j) = \{K_2 \mathbf{E}^2(i, j)\} / \{\sum_{i, j}^{HW} \mathbf{E}^2(i, j) + \sigma_2^2\}.$$

The resulting response $\hat{\mathbf{E}} \in \mathbb{R}^{H \times W \times 256}$ is interpreted as a post-stimulus time histogram and the motion energy is constrained to the same energy space as the local motion energy from stage I, as we further design an identical flow decoder to project the energy of each spatial location into the optical flow. The structure of the flow decoder is a nonlinear mapping process consisting of multiple 1×1 convolution blocks with residual connections, which are referred to in several

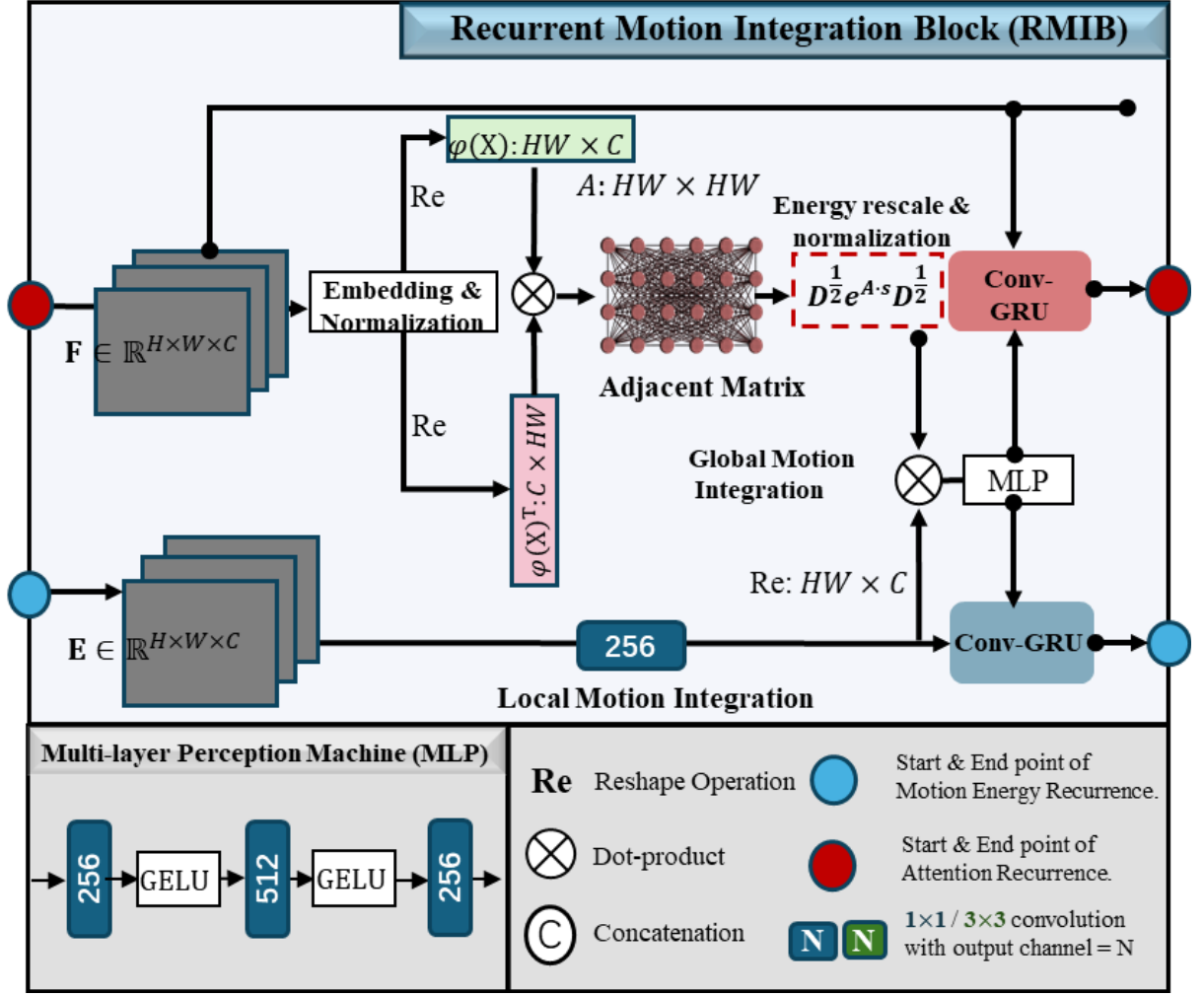


Figure 5.3: Molding of motion perception system (B): The global motion integration process based on a self-attention mechanism, named recurrent motion integration block (RMIB), a sub-block of Figure 5.2.

current advanced optical flow estimation models [87, 164]. Additionally, a convex upsampling strategy [171] is employed to restore the optical flow’s original resolution. The entire architecture of stage II is illustrated in Fig. 5.2 (A, B).

5.2.3 Training Strategy

Current methods for estimating optical flow using deep neural networks (DNNs) can be categorized into unsupervised/self-supervised and supervised learning approaches. While unsupervised learning methods intuitively mimic creatures’ interactions with the world, most current methods based on differentiable image warping [78, 111, 159] still attempt to approximate the physical motion ground truth (GT). Therefore, in this work, we primarily focus on the supervised learning approach, which is more straightforward as recent research suggests that human perception

of motion is reasonably similar to physical GT [189]. Our primary aim is to evaluate how well the model approximates human motion perception rather than its accuracy in predicting GT.

To train and evaluate the model, we constructed a dataset containing various natural and artificial motion scenes. Specifically, we incorporated the Sintel benchmark [18], the DAVIS dataset with pseudo-labels generated by FlowFormer [62], as well as self-created multi-frame datasets with non-texture motions and drifting grating motion. Including simple motion stimuli and drifting gratings allows the model to generalize under different non-texture conditions while providing a potential slow-world Bayesian prior [184]. The model is first pre-trained with simple motion and subsequently fine-tuned on complex natural scenes to facilitate convergence [68].

To ensure the model performs in a manner conceptually similar to how humans unsupervisedly learn motion estimation by watching natural videos, we also provide results from the unsupervised learning version. This version maximizes the reconstruction loss by warping image pairs based on intermediate optical flow. We incorporated dynamic scene augmentation and the self-distillation technique proposed in Chapter 3. The results of the unsupervised learning version will be discussed in Section 5.4, along with the supervised learning results. More specific training details can be found in the section 5.5.¹

5.3 Analyses

Fig. 5.4 shows the distribution of trained parameters in the first stage: (A) presents the distribution of velocity and orientation of the units; (B) displays the spatiotemporal frequency tuning of the units; (C) demonstrates that the complex cells in the first stage are capable of handling single-frequency component motions such as drifting gratings. The design of the trainable motion energy unit allows for optimal fitting to the flow statistics of the dataset. Although the distribution of the trained parameters appears to lack specific characteristics other than uniformity, it does reflect the effect of training. To validate the effectiveness of training the motion energy units, we conducted experiments with a fixed tuning parameter design using a uniform distribution sampled at equal intervals in terms of spatiotemporal frequency and orientation. The results showed that stage I, without the fitting function, significantly degraded the model’s ability to estimate motion. (See **Our-fixed** in Table 5.1 for details).

The following three parts are conducted for analysis: 1) In silico neurophysiological test of the activation pattern of the units; 2) Psychophysical stimulus tests comparing human perception and model response; 3) Natural scene test of the generalizability of the model in complicated scenarios.

5.3.1 In silico Neurophysiological Study

Directional Tuning: Some V1 and MT neurons respond selectively to a specific range of

¹The project code are publicly available on <https://huggingface.co/spaces/Zitang/Self-attention-based-V1MT-motion-model>

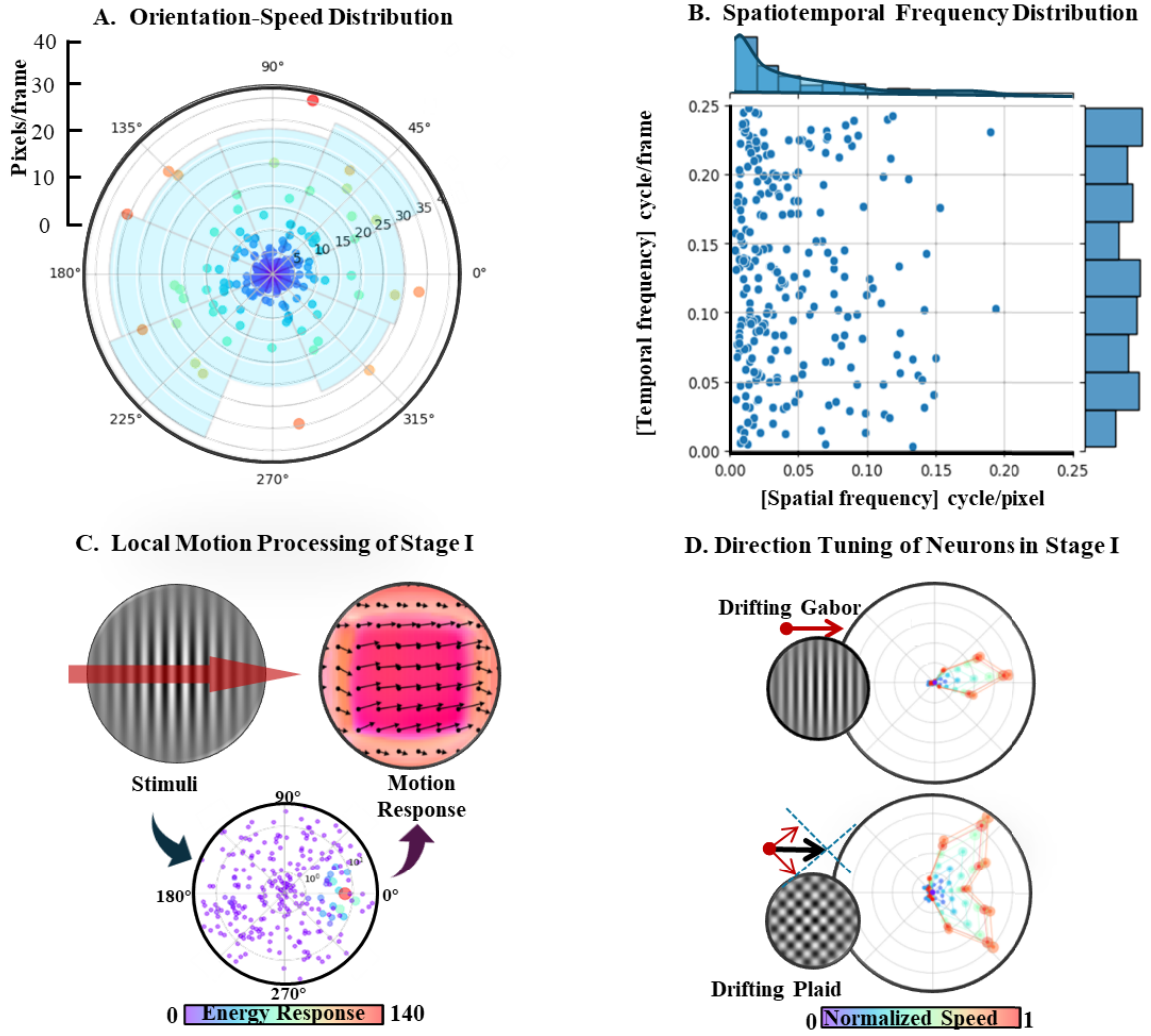


Figure 5.4: Properties of trained complex cells in stage I, where the ability to tune simple motions with a single spatiotemporal frequency component has emerged.

motion directions. To investigate the directional tuning of the units in our model, we utilized *in silico* neurophysiology to measure the activation patterns of 256 units in response to drifting Gabor and plaid stimuli. A plaid consists of two superimposed drifting Gabors, as illustrated in Fig. 5.5 (C). Analysis of the data revealed three distinct groups of units based on their partial correlation to Gabor and plaid stimuli: 1) Component cells, which always respond to the direction of a Gabor component; 2) Pattern cells, which respond to the integrated (coherent) motion direction of plaid; and 3) Unclassified cells, which do not show a clear preference for either component or pattern motions, as illustrated in Fig. 5.5 (C).

The distribution of these cell types is not uniform, with component cells being more commonly found in the primary visual cortex (V1), while pattern cells are more often observed in the MT and MST regions [117]. In agreement with this, as illustrated in Fig. 5.5 (A) (B), our

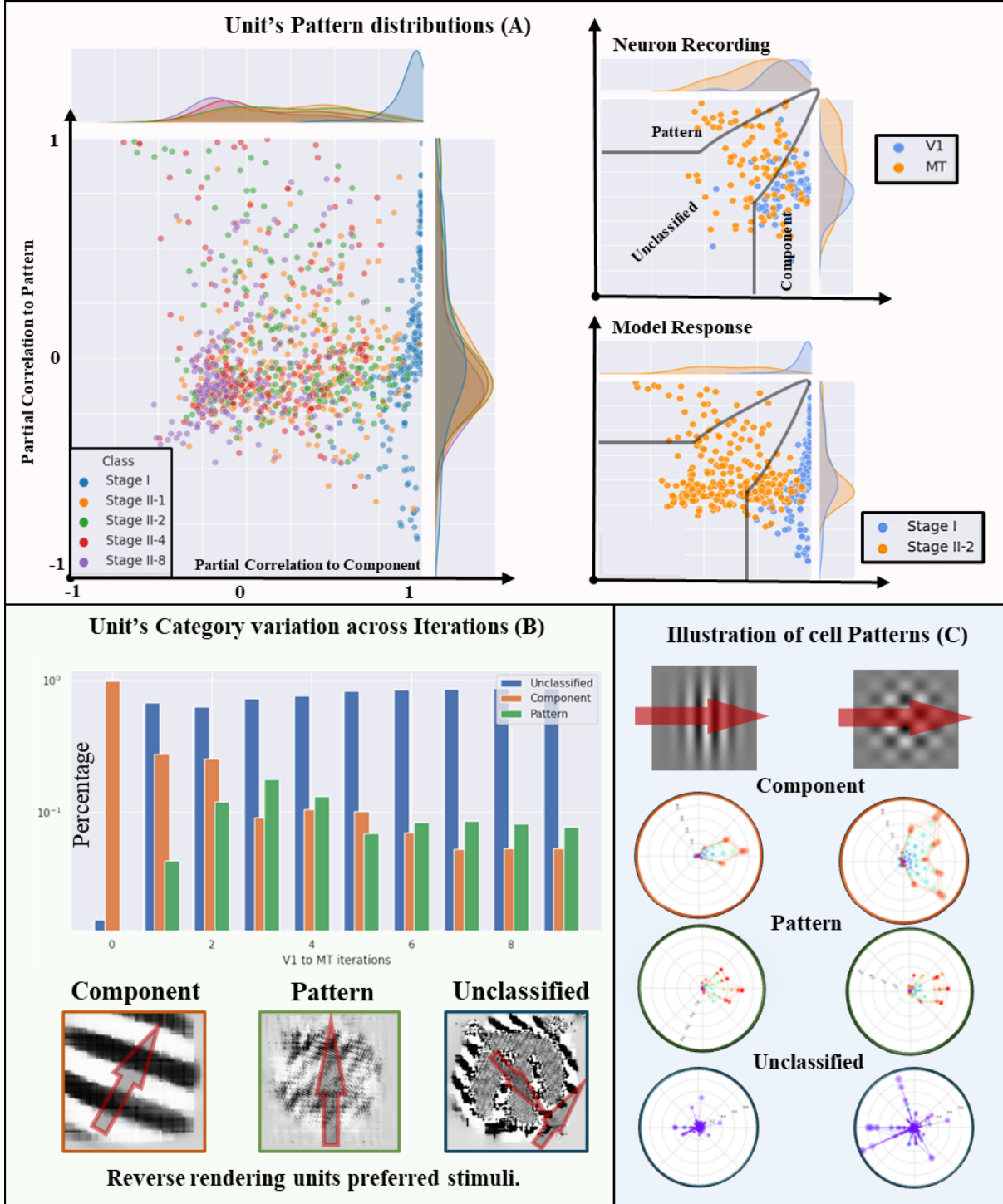


Figure 5.5: In silico neurophysiological test: directional tuning. (A, B, C): Analysis of the direction tuning of different groups of neurons for 1D drifting Gabor and 2D plaid stimuli; Stage II- N means the result from iteration N th in stage II; The neural distribution in (A) is redrawn from [117]. See section 5.5 for implementation details.

model demonstrates that in the first stage, most units tend to be component cells, whereas the number of pattern and unclassified cells increases in the second stage. In addition, we employed

the maximizing activation method [192] to reversely render their cellular preferences, showing that the second-stage unclassified units respond to more complicated motion patterns consisting of both central and background motions, as presented at the bottom of Fig. 5.5 (B). This suggests that the classical classification of motion neurons into component and pattern cells might be insufficient to characterize the motion integration properties of these neurons.

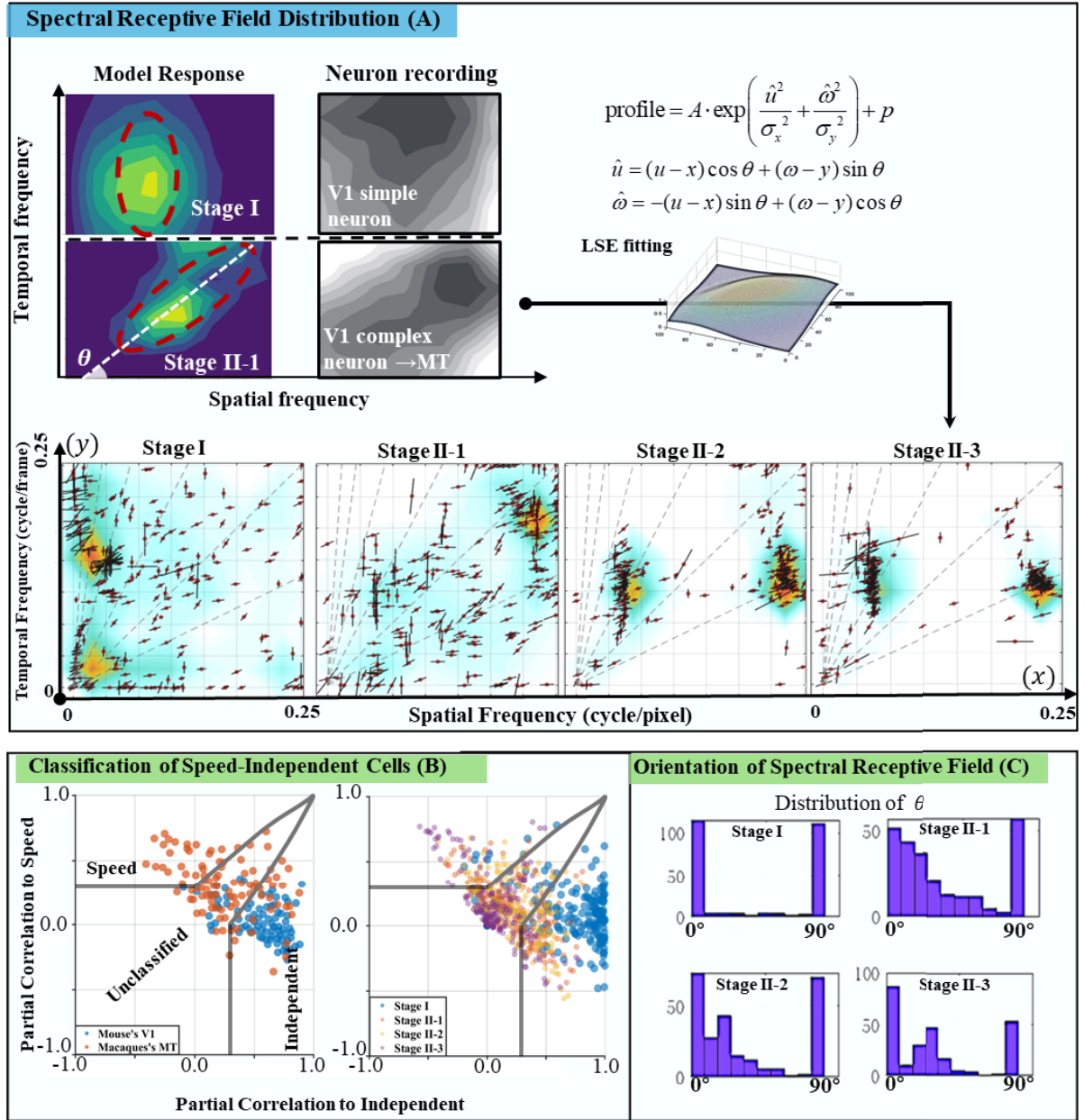


Figure 5.6: In silico neurophysiological test: speed tuning. Stage II- N means the result from iteration N th in stage II; (A, B, C): Analysis of the spectral receptive field and speed tuning of different groups of neurons. Neurons' spectral receptive fields in (A) are redrawn from [138]. Two neural distributions in (B) are separately taken from [85] and [136]. See section 5.5 for implementation details.

Spectral Receptive Field and Speed Tuning: We test the spectral (spatial-frequency-vs-temporal-frequency) receptive field of the model’s units using a combination of drifting Gabors with different spatiotemporal frequencies. Two dimensional oriented Gaussian contours are used to depict the receptive field of the 256 cells, fitted by minimizing the least square error. The results are shown in Fig 5.6 (A). Visually, the distribution of the receptive field tilt angle spreads from horizontal/vertical directions to oblique directions (Fig. 5.6, B). This indicates that the units in stage II have significant speed tuning compared to stage I. Speed tuning is a characteristic of higher-order visual motion neurons [135] and is often found in the MT area [137]. This tendency can be seen from the distribution of the partial correlation between the actual receptive field and its speed prediction/independent prediction [136], as demonstrated in Fig. 5.6 (C). Our two-stage process shows a degree of consistency with the change in mammalian neural distribution from the V1 to MT area.

5.3.2 Psychophysical Analysis

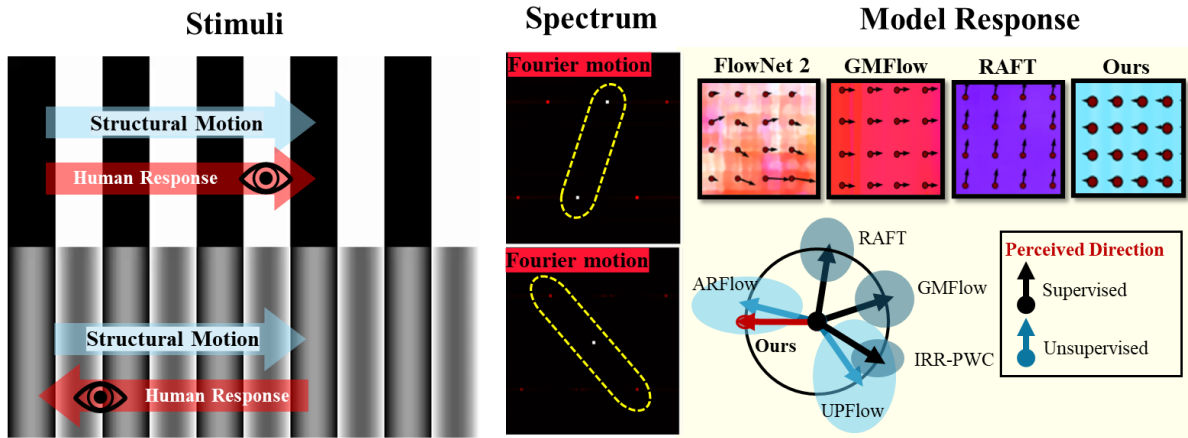


Figure 5.7: Missing fundamental illusion. By removing the base frequency of the Fourier series of the original square wave, the spectrum energy is flipped, and the perceived motion direction appears reversed. The size of the shaded area (right side) indicates the pixel-wise standard deviation (STD) of the response.

Fourier motion:

In the “missing fundamental illusion” [17], as shown in Fig. 5.7 (C), when the first spatial harmonic is removed from a square-wave grating with a quarter-cycle shift, the human perceived motion direction appears reversed.

This occurs because the dominant energy in Fourier space is oriented leftward, and human perception can be influenced by the Fourier components of the motion. Consider a square wave

with period T . The square wave $f(t)$ can be defined as:

$$f(t) = \begin{cases} 1 & \text{for } 0 \leq t < \frac{T}{2} \\ -1 & \text{for } \frac{T}{2} \leq t < T \end{cases}$$

The Fourier coefficients c_k of $f(t)$ are non-zero only for odd harmonics and can be calculated as:

$$c_k = \begin{cases} \frac{2}{ik\pi} & \text{if } k \text{ is odd} \\ 0 & \text{if } k \text{ is even} \end{cases}$$

If the signal $f(t)$ is shifted in time by $T/4$, i.e., $t_0 = \frac{T}{4}$, the Fourier coefficients will be multiplied by a phase factor:

$$f\left(t - \frac{T}{4}\right) \leftrightarrow c_k e^{-ik\omega_0 \frac{T}{4}} = c_k e^{-ik \frac{2\pi}{T} \frac{T}{4}} = c_k e^{-ik \frac{\pi}{2}}$$

Therefore, the Fourier series of the square wave with a time shift of $T/4$ is:

$$f\left(t - \frac{T}{4}\right) = \sum_{k \text{ odd}} \left(\frac{2}{ik\pi} e^{-ik \frac{\pi}{2}} \right) e^{ik\omega_0 t} \quad (5.7)$$

$$= \frac{2}{\pi} \left(\cos(\omega_0 t) - \frac{1}{3} \cos(3\omega_0 t) + \frac{1}{5} \cos(5\omega_0 t) - \dots \right) \quad (5.8)$$

where $\omega_0 = \frac{2\pi}{T}$ is the fundamental frequency. When the first harmonic $\cos(\omega_0 t)$ is removed, the remaining dominant energy becomes negative, corresponding to motion in the opposite direction.

Our model, whose first stage estimates motion from the Fourier components [25], can predict the reversed moving direction, similar to human response. In contrast, computer vision models like RAFT, GMFLOW, and FlowNet, designed to infer optical flow based on structural correspondence, either demonstrate poor generalization ability to these stimuli or do not exhibit motion bias from the Fourier component. These experimental results indicate that human motion perception is not solely based on tracking the object's correspondence.

Mario Reversed-Phi Illusion [190]: This stimulus demonstrated the reverse-phi illusion in psychology, where the subject perceives continuous motion despite the absence of actual motion. Current DNN models provide unstable results for this kind of motion (see [190] for the result of FlowNet2.0 [68]), while our model produces the illusion of motion similar to the continuous motion perceived by humans; check Figure 5.9 for our result.

Motion Integration: The direction of 1D motion stimuli, such as drifting Gabors, is ambiguous due to the aperture problem. When presented alone, they are perceived to move in orthogonal directions of stripes. When superimposed with other 1D motion components in different orientations, 2D directions consistent with both components are perceived. Pattern motion neurons in MT may contribute to this phenomenon. The second stage of our model can explain this as well, as shown in Fig. 5.8. Psychophysics also demonstrates motion integration across space. Fig. 5.8 (A) shows a psychophysical stimulus consisting of drifting Gabors [4]. These Gabors have a variety of local directions and speeds, yet all of them are consistent with

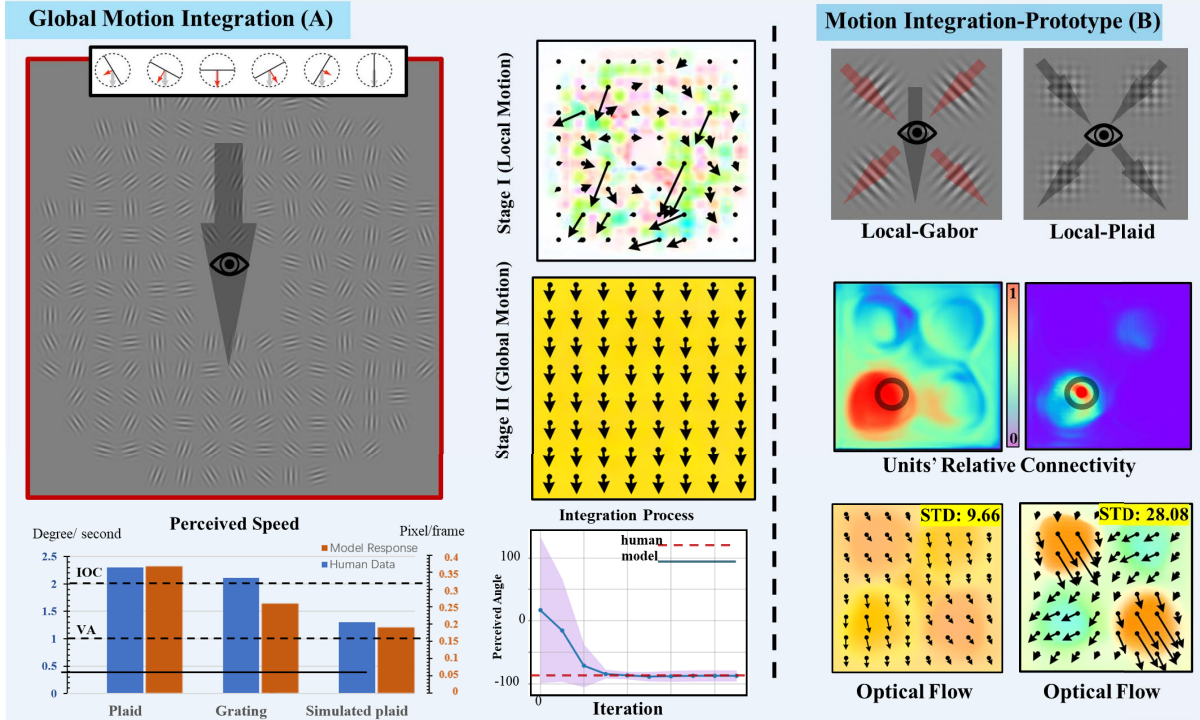


Figure 5.8: Psychophysical stimulus tests motion integration. (A, B) : Spatial integration of 1D and 2D motions. In the stimulus panels, grey arrows indicate the direction perceived by humans, while red arrows indicate the direction of physical motion. The middle panel of (B) : is the heat map representing the neuron's connectivity from a selected local region (circle center) to other regions, with the warmer color indicating a stronger connection. The human psychophysical data in (A), were redrawn from the [4].

one global 2D motion (downward in this case). When viewed as a whole, humans do perceive coherent downward motion. Stage II of our model predicts both the perceived direction and speed of the global Gabor motion.

Fig. 5.8 (B) compares spatial motion integration between 1D Gabor motion (left) and 2D plaid motion (right). Humans are able to perceive global downward motion only in the former case: The heat maps depict how units with high activity establish long-distance connections to resolve the aperture problem when subjected to Gabor (ambiguous motion) stimuli. In contrast, the plaid stimuli (determined motion) suppress these long-distance connections. In the latter case, local integration of motion signals takes priority over global integration. Once the local ambiguity is resolved, the global integration process is suppressed. Our model can predict such adaptive motion pooling in human visual processing [4].

Furthermore, the barber pole illusion demonstrates how locally ambiguous 1D motion is affected by the shape of the moving area [166]. Specifically, as the height-width ratio of the visual area varies, human perception of direction shifts from oblique to vertical [45] (Fig. 5.10).

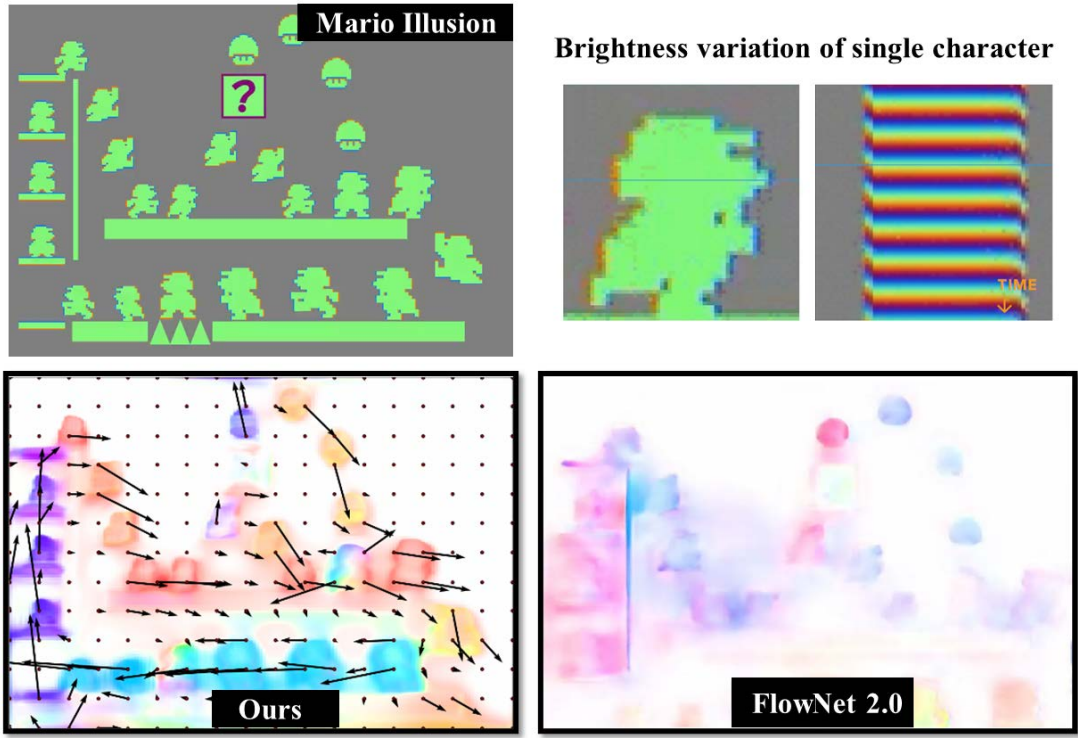


Figure 5.9: Mario reverse-phi illusion. Stimuli is adopted from [190].

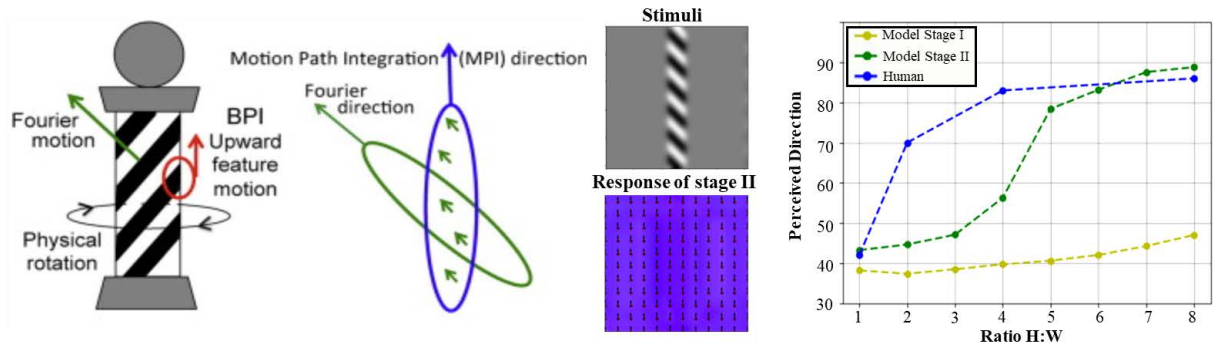


Figure 5.10: Barber pole illusion. The human psychophysical data were redrawn from the [45].

Our model can predict the shift in perceived direction in stage II, showing its ability to integrate motion signals with boundary orientations. For more video demonstrations, such as the reverse phi illusion [6], please see section 5.5 for details.

Comparison of Complex Natural Scenes: The proposed model can effectively handle natural scenes, as demonstrated in Fig. 5.11 (A). Natural stimuli contain diverse spatiotemporal frequency components, leading to complex activation patterns in Stage I. From the decoded flow field, the motion of stage I is limited to localized areas. For example, local motion cannot be found in untextured road areas. This situation necessitates long-range interactions with the surrounding spatial context, which our recurrent integration process in stage II effectively

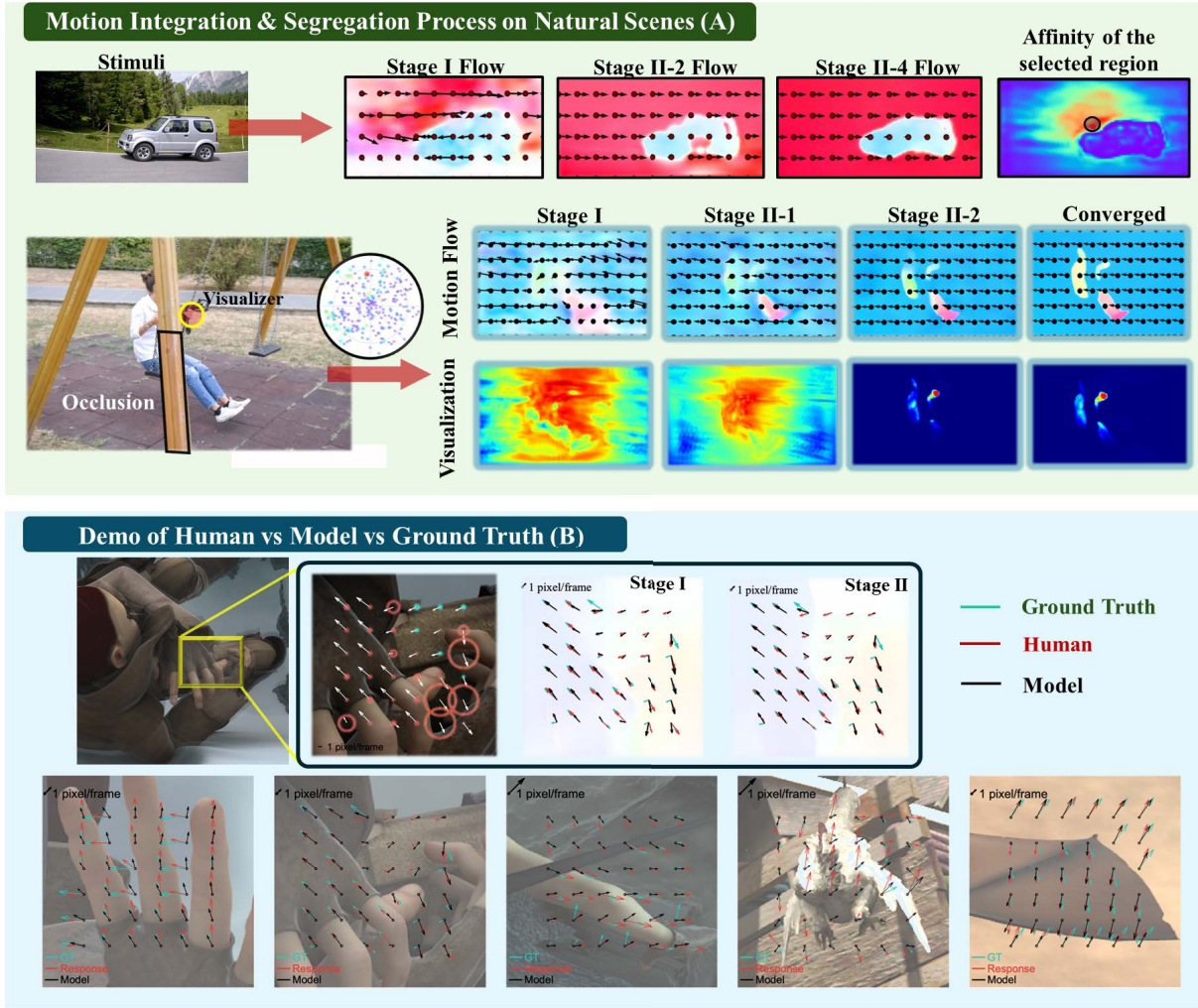


Figure 5.11: Performance on complex natural scenes. **(A)**: Our model can estimate a dense natural optical flow with recurrent integration. “Stage II-N” refers to the outcomes from the Nth iteration in Stage II, with visualizations illustrating neuron connectivity via heat maps. These maps highlight connections from a selected local area (Visualizer) to other regions **(B)**: Similarity of our model estimation to human perceived flow on the Sintel dataset. larger red circles show a closer response to humans over the GT, while blue circles indicate the reverse. Despite being trained to fit the GT, our the model uniquely exhibits a higher correlation with human responses than with the GT.

accomplishes. It is evident from the affinity heat map (right side of subfigure A) that object and background areas are clearly segregated in stage II. This suggests that the integration mechanism based on attention has the potential to combine motion integration and object segmentation into a single framework. These two processes are considered highly relevant in the human visual system [56]. In the bottom scene, the affinity heat map in stage II clearly distinguishes different regions, such as separating the girl from the background and adaptively building connections

over occlusions.

We used psychophysically measured optical flow from the Sintel dataset [189] as a benchmark for naturalistic scene flow perceived by humans. We first show the general distribution of our model’s response and human data, illustrated by a polar map 5.12 (A) and (u,v) vector map (C). Globally, the model well captured the human’s response with a high deceptive ability of $R = 0.81$.

Further, Our model was compared to several optical flow estimation methods used in computer vision, including classical algorithms like Farneback, as well as SOTA DNN-based models. We evaluated officially released DNN models that utilize a wide range of inference structures, such as multi-scale inference [38, 68], spatial recurrent models [171], graph reasoning [98], and vision transformers [187].

As shown in Table 5.1, we computed both the Pearson correlation coefficients and vector endpoint errors (EPEs) between the model prediction and human response, or ground truth (GT). Additionally, we examined the partial correlation between humans and models while controlling the impact of GT:

$$\rho_{\text{model}} = r_{\text{resp model} \cdot \text{GT}} = \frac{r_{\text{resp model}} - r_{\text{resp GT}} \cdot r_{\text{model GT}}}{\sqrt{1 - r_{\text{resp GT}}^2} \sqrt{1 - r_{\text{model GT}}^2}}, \quad (5.9)$$

where r is the Pearson correlation. This measure is critical in validating the models’ ability to capture the characteristics of the human response, as any model could appear to have a high correlation with the human response by just approximating the GT due to a high correlation between the human response and GT [189].

Quantitatively, the proposed model outperformed all compared models in terms of partial correlation. The RAFT-val in the table is the RAFT framework trained on our dataset as validation and indicates that our mixed training set also improves the explanatory power of the human response. Fig. 5.12 (B) shows that our model significantly improves the partial correlation across iterations in Stage II, indicating that the proposed recurrent motion integration architecture can generate more human-like deviations from GT, which is a trend not present in a similar recurrent network, RAFT. In Fig. 5.11 (D), one can directly see that our model prediction is more similar to human-perceived flow than the GT flow.

Table 5.1 also shows the results of three biologically-inspired models. FFV1MT [155] is a model capable of computing dense optical flow using direct decoding of the Simoncelli & Heeger V1-MT mechanism. The other two models are modified versions of MotionNet [144] and DorsalNet [115], recently proposed DNN-based models for the explanation of neural responses to visual motion stimuli. Since the original models are designed to recognize global motion only, we tested general multi-layer 3D CNN (a core component of MotionNet and DorsalNet) with residual connections, and a pre-trained DorsalNet with frozen parameters, with a linear flow decoder to compute dense flow, trained on natural dense optical flow datasets. Our model outperforms these biologically-inspired models in predicting human responses. The low performances of

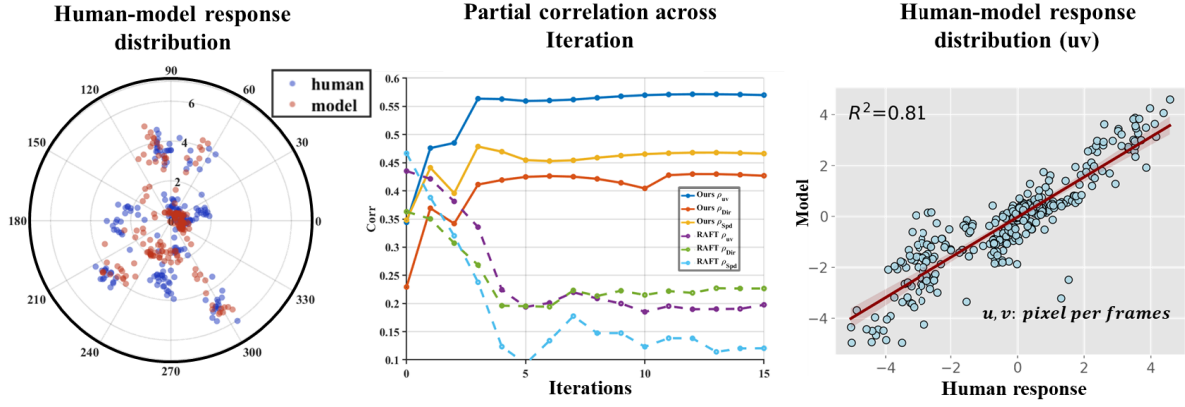


Figure 5.12: Comparison of our model’s motion estimations with human-perceived motion on the Sintel Benchmark.

Table 5.1: Model v.s. Human v.s. Ground truth. ρ : Partial correlation between human & model controlling GT; r : Pearson correlation coefficient; epe : vector end-point error; uv , dir , spd represent motion components in Cartesian space, direction, and speed, respectively.

Method	ρ_{uv}	ρ_{dir}	ρ_{spd}	v.s. Human				v.s. GT			
				r_{uv}	r_{spd}	r_{dir}	epe	r_{uv}	r_{spd}	r_{dir}	epe
Farneback [42]	0.27	0.23	0.11	0.41	0.91	0.34	2.02	0.34	0.33	0.92	1.96
FlowNet2.0 [38]	0.39	0.26	0.34	0.92	0.90	0.96	0.94	0.95	0.94	0.98	0.47
RAFT [171]	0.20	0.22	0.14	0.92	0.90	0.96	0.93	0.98	0.99	0.99	0.25
RAFT-val	0.43	0.17	0.42	0.92	0.89	0.96	1.01	0.92	0.89	0.98	0.69
AGFlow [98]	0.30	0.16	0.20	0.93	0.90	0.96	0.92	0.98	0.98	0.98	0.27
GMFlow [187]	0.34	0.32	0.17	0.91	0.84	0.96	1.03	0.93	0.90	0.97	0.73
FlowFormer [62]	0.36	0.14	0.32	0.93	0.91	0.95	0.90	0.98	0.97	0.98	0.42
FFV1MT [155]	0.31	0.16	0.31	0.83	0.64	0.92	1.48	0.59	0.84	0.94	1.29
3DCNN	0.27	0.29	0.42	0.83	0.86	0.95	1.31	0.83	0.86	0.96	1.14
DorsalNet [115]	0.17	0.19	-0.10	0.20	-0.08	0.86	2.35	0.20	-0.04	0.86	2.33
Ours-fixed	-0.02	0.12	0.16	0.31	0.23	0.78	2.24	0.35	0.18	0.80	2.29
Ours-Stage I	0.34	0.23	0.35	0.71	0.71	0.92	1.52	0.67	0.67	0.92	1.49
Ours-Stage II	0.57	0.43	0.47	0.91	0.88	0.95	0.98	0.86	0.87	0.95	1.04

the FFV1MT model and the modified DorsalNet also suggest that accurately estimating dense optical flow is a challenging task, requiring specific design considerations to address complex and long-range spatial interactions, large jumps, and boundary effects, the complexities of which are not adequately captured by simple mechanisms.

Furthermore, we provide a qualitative demonstration of our model against other related motion models. As shown in Figure 5.13, we compare three relevant models: a pre-trained

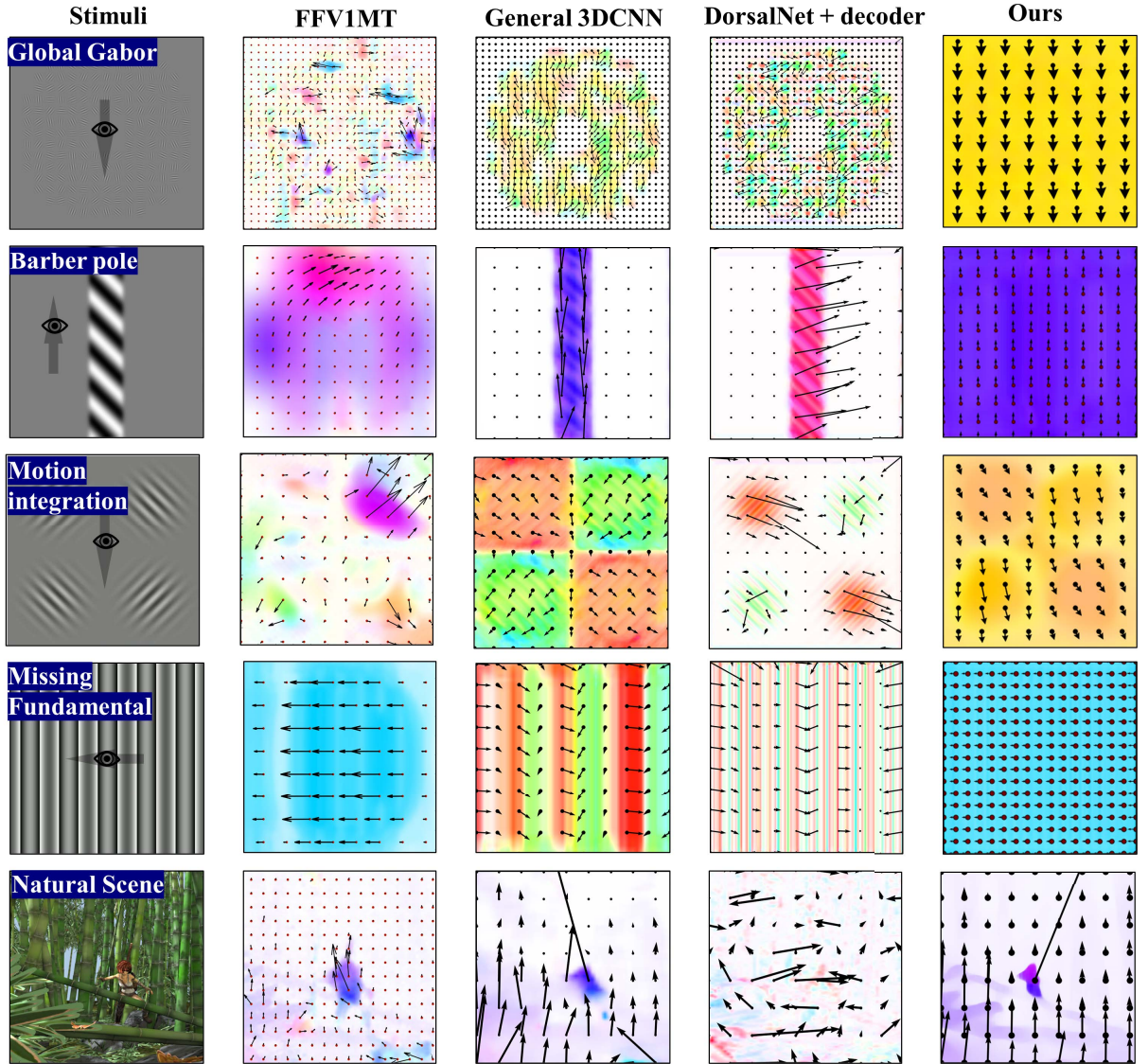


Figure 5.13: Qualitative comparison between our model and other related models. Please refer to the video in the 5.5 for a detailed view of the stimuli used. FFV1MT: estimating dense optical flow with simple decoding of the Simoncelli & Heeger V1-MT mechanism. General 3DCNN: multi-layer stacked 3DCNN with residual connection, trained on natural dense optical flow dataset. DorsalNet+decoder: pre-trained Dorsalnet (freeze param) with one layer of 3D convolution as a linear flow decoder, decoder trained on natural dense optical flow dataset.

DorsalNet with a linear flow decoder, a general 3D CNN (the basic structure for MotionNet and DorsalNet), and FFV1MT [155], the only model known to calculate dense optical flow with simple decoding of the Simoncelli & Heeger V1-MT mechanism. These models fall short in terms of dense optical flow estimation accuracy or fail to account for the global integration of local motions. This outcome suggests that a simple linear decoder cannot handle dense optical flow, particularly in complex real scenes. In addition to flexible spatial pooling and accurate

estimation of dense optical flow, our model consistently displays human-perceived Fourier motion in missing fundamental stimuli through motion energy computation simulating V1 neurons. The comparison with other models highlights the difficulty of achieving such diverse capabilities **simultaneously** in a single model.

5.4 Discussion on Unsupervised Learning

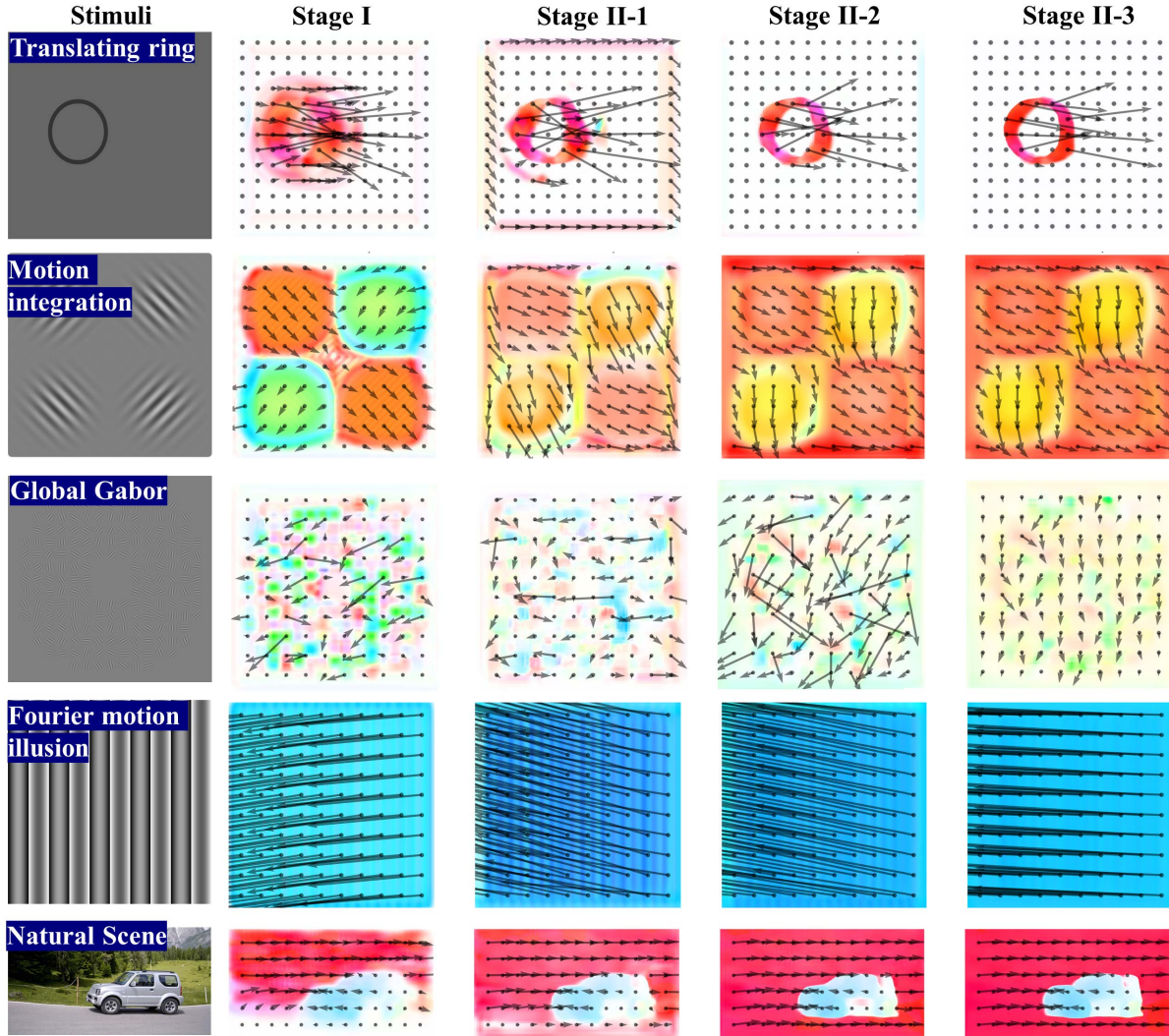


Figure 5.14: Qualitative result of unsupervised learning methods. We incorporated the unsupervised learning method proposed in Chapter 3 into our model and trained it on natural videos. The results indicate that unsupervised learning methods can replicate many characteristics of supervised learning, such as the missing fundamental illusion, and are capable of handling natural scenes. However, the unsupervised model exhibits weaker motion integration abilities compared to its supervised counterpart. Moreover, unsupervised learning is more sensitive to variations in luminance, such as moving shadows from vehicles, which will also be recognized as motion.

Animals spontaneously develop motion perception abilities by interacting with dynamic environments, a behavior akin to the concept of unsupervised learning. To imbue our human-oriented model with more natural learning behaviors, we incorporate the unsupervised learning method proposed in Chapter 3 and train it on natural videos. The core method is based on image-warping, which involves inputting two consecutive frames into a model that predicts an initial optical flow field. This predicted flow is then used to warp the second frame to align with the first frame. The difference between the original first frame and the warped second frame, known as the reconstruction loss, is calculated, and the model is optimized by minimizing this loss, thus learning to estimate optical flow without the need for labeled data. In addition to the basic multi-scale reconstruction loss, we employ dynamic scene augmentation methods, enabling the model to learn motion patterns in multi-frame dynamic environments.

We present the results of the unsupervised version in Figure 5.14. Generally, unsupervised learning can drive the model to recognize motion and handle various scenarios, such as Fourier motion and natural scenes, similar to supervised learning. The primary difference between supervised and unsupervised learning lies in motion integration ability. Although the unsupervised learning method demonstrates some integration capability, it is significantly weaker than the supervised one, as illustrated by the motion integration of global Gabor stimuli between Figure 5.8 and 5.14. Additionally, unsupervised learning is more affected by pixel luminance shifts, as its reconstruction loss function mainly relies on photometric loss. This can sometimes misinterpret luminance shifts as object-level motion, such as the movement of the shadow of the car in the bottom row. Thus, a more sophisticated strategy is needed for unsupervised learning methods in AI agents to achieve human-like perception by interacting with natural environments.

5.5 Methods and Implementation details

This section includes some of the implementation information and methods used in the study. Furthermore, to enhance the visual representation, we have created videos showcasing select stimuli. ²

5.5.1 Dataset

Current methods for estimating optical flow using DNNs can be categorized as unsupervised/self-supervised and supervised learning approaches. While unsupervised learning methods are intuitively similar to creatures' interaction with the world, most current methods based on differentiable image warping [78, 111, 159] still attempt to approximate physical motion GT. Therefore, we choose to use the supervised learning approach, which is more straightforward as recent research suggests that human perception of motion is highly similar to physical GT [189].

²The supplementary video demo can be found in https://github.com/KUCognitiveInformaticsLab/VideoDemo_Zitang/tree/main/chapter%205

To train and evaluate the model, we construct a dataset containing various natural and artificial motion scenes. Specifically, we incorporate the Sintel benchmark [18], the DAVIS [133] dataset with pseudo-labels generated by FlowFormer [62], as well as self-created multi-frame datasets with non-texture motions and drifting grating motion.

Simple non-texture motion: As the name suggests, non-texture motion refers to scenarios where both the moving object and the background exhibit low texture density. We incorporate non-texture motion as part of our training data due to the incompleteness of existing optical flow datasets. Our observations show that current DNN models exhibit suboptimal generalization performance on simple stimuli commonly employed in vision research, such as a drifting grating and a simple moving box. (Check Figure 4.2).

There are various explanations for this phenomenon. First, non-texture stimuli, such as drifting gratings, present multiple ambiguous motion solutions, which may confuse the model. Additionally, the model may not have been trained on non-texture datasets, leading to difficulties in generalizing to such stimuli. Consequently, the model’s instability causes significant deviations from human interpretation in such pure experimental scenarios, hindering the study of the model using stimuli commonly employed in vision science.

To alleviate the problem, we used a concise method to construct a non-texture dataset. We first employed a superpixel algorithm [1] to extract the self-regions of natural images, followed by a low-pass filter to smooth out the texture. We practiced affine transformations to simulate two-dimensional translations, rotations, deformations, etc., of the object and inverse affine transformations to track the optical flow. The motion of each step is sampled from a random Gaussian distribution, and we use a Markovian stochastic process to simulate inter-frame smoothed motion over multiple frames, where the current moment’s motion is correlated with the previous motion state. Specifically, we assume that random object movement $\mathbf{S}(t)$ can be decomposed into two orthogonal vectors $\mathbf{S}(t) = [U(t), V(t)]$. For any time step t , we introduce Markov properties by assuming that the motion state depends only on the motion in the previous time step $t - 1$:

$$\Pr[\mathbf{S}(t)] = \Pr[\mathbf{S}(t) = s_t \mid \mathbf{S}(t-1) = s_{t-1}] \quad (5.10)$$

The motion states $[U, V]$ at time t are sampled from the 2D Gaussian distributions with condition of previous motion state. Check Figure 4.2 for the demo of the non-texture training sample.

Drifting Grating: Drifting gratings are commonly used stimuli in visual studies, which contain only a single spatiotemporal frequency component. We used five different combinations of frequencies linearly sampled from $(0, \frac{1}{16}]$ (pixel/ frame), four different temporal frequencies linearly sampled from $(0, \frac{1}{8}]$ (pixel/ frame), and eight different orientations sampled uniformly between $(0, 2\pi]$. In total, there are 160 scenes, and each contains 32 continuous frames. Due to the aperture problem, there is no single solution for this class of motion, but we defined the labels with the slowest moving speed as perpendicular to the spatial stripe, which is most similar to the human interpretation. Incorporating this class of datasets into the training provides a

potential slow-world Bayesian prior of humans [184] to solve the ambiguous motion. Check Figure 4.2 for the demo of the training sample.

5.5.2 Training

Environment: For the model training, we implemented our method using the PyTorch 2.0 framework on a workstation equipped with four parallel Nvidia RTX A6000 GPUs, running on the CUDA 11.7 runtime. The data analysis was conducted using MATLAB, as well as Python packages such as SciPy and Pandas.

Timing Setting: Considering the current mainstream playback frame rate (25 FPS) and the time of the human visual impulse response (about 200 ms), we set the temporal window of the first stage to 6 frames (spanning 200 ms). In training, 11 consecutive images (400 ms) are fed into the model. The instantaneous velocity in the middle of the image sequence, i.e., the fifth frame, is used as the label for supervised training. The MSE loss was used to supervise the output from each iteration from Stage I to Stage II:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2$$

Hyperparameters: The total trainable parameters of the model is 14.7 M. During training, we set the iteration as eight in stage II. A total of nine flow fields are generated per inference, and we employ a sequence loss based on mean square error [171] to measure the difference between the predicted optical flow and the GT flow. The model is first pre-trained with simple motion and subsequently fine-tuned on complex natural scenes to facilitate convergence [68]. Specifically, we first train the model on simple no-texture motion and drifting grating for 150 epochs with a learning rate (lr) of $1e^{-4}$, and then fine-tune it on the Sintel and DAVIS dataset for an extra 150 epochs with a learning rate of $0.5e^{-4}$. The entire training process was optimized using the Adam optimizer [81] with linear decay of the learning rate and a batch size of 24. Mixed precision training and gradient clipping techniques were employed in all training stages for faster convergence. For image preprocessing, all images were resized to 448×960 and then randomly cropped into 384×576 . We also adopted random image horizontal flipping for data augmentation.

5.5.3 In Silico Neurophysiological Test

We utilized drifting Gabor or plaid (superimposed by two Gabor components) with a single frequency component as the input stimulus (Check Figure 2.7).

Referring to Figure 5.1 (D), the model’s response value at Stage I, or after the remap operation of each iteration in Stage II, was considered as the poststimulus time histogram of the unit. We averaged the responses across spatial dimensions using a Gaussian weighted window to

obtain the activation distribution of the total of 256 units, i.e., $\mathbb{R}^{1 \times 1 \times 256}$, with respect to input stimulus. The image sequence is generally set to 512×512 with full contrast.

Directional tuning

We used a single frequency drifting-Gabor and plaid (superimposed by ± 30 degrees) as the stimulus input. First, we selected 12 directions uniformly sampled from $(0, 2\pi]$. For each direction, we linearly sampled $8 \times 8 = 64$ sets of spatiotemporal frequency combinations and thus obtained 64 directional tuning curves for each unit using the drifting-Gabor stimulus. We selected the spatiotemporal frequency with the largest tuning standard deviation as the preferred frequency st^* of each unit. Then, we input the Gabor and plaid with the frequency configuration of st^* into the model and obtained the direction tuning curve of each unit. Denoting the model's tuning curve on the drifting-Gabor with st^* as $\mathcal{C}$ and the tuning curve on the plaid with st^* as $\mathcal{P}$, we calculated the directional tuning capability of each cell using a pair of partial correlations:

$$R_{\text{pattern}} = \frac{(r_p - r_c r_{cp})}{\sqrt{((1 - r_c^2)(1 - r_{cp}^2))}}, \quad R_{\text{component}} = \frac{(r_c - r_p r_{cp})}{\sqrt{((1 - r_p^2)(1 - r_{cp}^2))}}, \quad (5.11)$$

where r_c is the correlation of the $\mathcal{P}$ with the component prediction, which is generated by the superimposed ± 30 -degree shift of the $\mathcal{C}$; r_p is the correlation of the $\mathcal{P}$ with the pattern prediction, which is equal to $\mathcal{C}$; and r_{cp} is the correlation between the two predictions. We labeled units as "component" if the component correlation coefficient significantly exceeded either 0 or the pattern correlation coefficient, whichever was larger. Similarly, we labeled units as "pattern" if the pattern correlation coefficient significantly exceeded either 0 or the component correlation coefficient. Units were labeled as "unclassified" if both pattern and component correlations significantly exceeded 0 but did not differ significantly from one another, or if neither correlation coefficient differed significantly from 0. The visualized result is plotted in Figure 5.5

We employed maximum activation technology [192] to render the unit's preferred motion stimuli reversely. The initialized motion stimuli are composed of spatiotemporally incoherent Gaussian white noise $I \in \mathbb{R}^{H \times W \times T}$, which is continuously updated by a gradient ascent method to maximize the activation response of each unit. The activation response of each unit is obtained by a weighted average pooling of its responses $a \in \mathbb{R}^{H \times W}$ across all spatial locations. To consider the continuity of the motion, we add temporal smoothing and spatial smoothing regular terms based on the first-order gradient to the optimization function, which is defined as:

$$\mathcal{A}(I) = \sum_{\Omega} \mathcal{G}_i a_i + w \left(\left| \frac{\partial I(x, y, t)}{\partial x_1} \right| + \left| \frac{\partial I(x, y, t)}{\partial y} \right| + \left| \frac{\partial I(x, y, t)}{\partial T} \right| \right); \quad (5.12)$$

where Ω denotes the set across the image space and $\mathcal{G}$ is a 2D Gaussian kernel defined on the Ω . We iterate $I := I + w \frac{\partial}{\partial I} \mathcal{A}(I)$ so as to maximize the unit's activation. Each unit went through approximately 300 iterations to generate the following stimuli:

1. For component units (Stage I), which prefer motion components of a single spatiotemporal frequency.

2. For pattern units (Stage II-1), which prefer textures or plaids that move in a specific direction.
3. For unclassified units (Stage II-4), which have a variety of complex motion formats that go beyond the interpretation of simple directional tuning.

Spectral Receptive Field and Speed Tuning

To investigate the spectral properties of the units, we used a combination of drifting-Gabor stimuli with different spatiotemporal frequency components. We first sampled 16 directions uniformly from the interval $(0, 2\pi]$. For each direction, we tested $10 \times 10 = 100$ spatiotemporal frequency combinations, with 10 logarithmically spaced spatial frequencies between 0.004 and 0.125 pixels/cycle and 10 logarithmically spaced temporal frequencies between 0.04 and 0.25 pixels/frame. Then we selected the direction with the largest sum of activations across all 100 spatiotemporal combinations as the preferred motion direction $\mathcal{D}^*$. Each unit under its $\mathcal{D}^*$ has 10×10 responses under different spatiotemporal combinations, constituting a grid in logarithmic coordinates. This grid of responses is visualized on the bottom left side of Figure 5.6 (D) and is referred to as the spectral receptive field of each unit.

Following [135], we use a 2D oriented Gaussian profile to fit the spectral receptive field of each unit, where the 2D Gaussian is defined as:

$$\mathcal{P} = A \cdot \exp\left(\frac{\hat{u}^2}{\sigma_x^2} + \frac{\hat{\omega}^2}{\sigma_y^2}\right) + p, \text{ s.t. } \begin{cases} \hat{u} = (u - x) \cos \theta + (\omega - y) \sin \theta \\ \hat{\omega} = -(u - x) \sin \theta + (\omega - y) \cos \theta \end{cases} \quad (5.13)$$

The Matlab `fmincon` function is used to search the best fitting based on the LSE loss function within $\{(\theta, \sigma_x, \sigma_y, x, y) \mid \theta \in [0, \frac{\pi}{2}); \sigma_{x,y} \in (0, \infty); x, y \in [-1, 1]\}$:

$$\ell_{lse} = \|\mathcal{P} - RF\| + w \frac{\sin(2\theta)}{\exp(\frac{|\sigma_1 - \sigma_2|}{\sqrt{\sigma_1 \times \sigma_2}})}, \quad (5.14)$$

where the second term is a weighted regularization to prevent tilting (i.e., $0 < \theta < 90$) when the receptive field is close to a circle. The fitting results are illustrated in Figure 5.6 (A), where each red dot represents the midpoint of the receptive field and the slope of the bar indicates the orientation of the receptive field, which is consistent with the visualization style in [135].

We further employed a pair of partial correlations to validate the speed-tuning capability for each unit [137]:

$$R_{\text{indep}} = \frac{(r_i - r_s r_{is})}{\sqrt{(1 - r_s^2)(1 - r_{is}^2)}}, \quad R_{\text{speed}} = \frac{(r_s - r_i r_{is})}{\sqrt{(1 - r_i^2)(1 - r_{is}^2)}} \quad (5.15)$$

where R_{indep} and R_{speed} are the partial correlations of the response filed with the independent and speed-tuned predictions, r_i is the correlation of the data with the independent prediction, r_s is the correlation of the data with the speed-tuned prediction, and r_{is} is the correlation of the two predictions, as shown in Figure 5.6 (B). Here, the spatiotemporal-frequency-independent

prediction is computed by taking the outer product of the two 1D tuning curves, and the speed-tuned prediction is computed by shifting the temporal frequency as a function of spatial frequency so that the preferred speed is independent of the speed tuning of the unit. (See [137]’s Figure 5 for details).

5.6 Discussion and Conclusion

DNNs have achieved impressive performance in various vision tasks, and their ability to explain the HVS is an active area of research. Recent studies have employed DNNs to model and understand the neural mechanism of visual motion. For instance, Rideaux et al. [143, 144] and Nakamura and Gomi [118] used multilayer feedforward networks, while Storrs et al. [161] used a predictive coding network (PredNet), to model biological visual motion processing, and found similarities to neurophysiological data. De Jong et al. [32] found that the spatiotemporal frequency tuning properties of some units in FlowNet resemble those found in mammalian neurons. DorsalNet [115] uses first-person perspective video stimuli to train a 3D ResNet model to predict self-motion parameters, which helps the model recapitulate the neural representation of the dorsal visual stream.

With a similar goal in mind, this Chapter simplified the biological motion process pipeline and proposed a two-stage architecture that models the complete pathway from images, through neural representations, to the perceptual response. Through end-to-end training using a wide range of datasets, our model generalizes well from simple stimuli to complex natural scenes and partially captures important characteristics of motion-processing neurons, including a change in spatiotemporal tuning from V1 to MT areas. To model the motion integration function, we introduced a novel recurrent process based on the attention mechanism. This process successfully explains a wide range of physiological findings (e.g., a change in the population of component and pattern cells from V1 to MT) and psychophysical findings (e.g., global motion pooling). It also improves the partial correlation with human psychophysical response. The success of the attention mechanism in motion integration could be attributed to its similarity to the human visual grouping mechanism [109] or similar feature grouping/binding that may be accomplished using a top-down attentional selection mechanism [175].

While we show that the attention-based recurrent network is a promising computational model of human visual motion grouping and segmentation, how its complex architecture (where all responses can be influenced by all other responses) is actually implemented in the human brain remains an open question. Another limitation of the current model is that it does not process several important abilities of human motion perception, such as non-Fourier (second-order) motion detection [25], which will be further discussed in Chapter 6. Our model does not take into account the benefits of biological processing, such as energy efficiency. In conclusion, combining classical motion energy and advanced deep learning technology is a promising approach to bridge the gap between human and DNN motion perception systems. The proposed architecture and

recurrent process offer insights into the underlying mechanisms of motion perception and open up new avenues for future research.

Chapter 6

Human-aligned Visual Motion Processing Model: Second-order Motion

In the previous Chapter 5, we built a human-aligned visual motion processing model. This model, based on motion energy computation, efficiently captures motion based on the local brightness of spatial patterns, defined as first-order motion.

Nonetheless, many animals, including flies, cats, and humans, can visually recognize image motion not only from shifts of spatial patterns defined by luminance modulations but also from those defined by high-level image features such as temporal modulations and contrast modulations (second-order motion). The model proposed in Chapter 5 is theoretically not applicable to such second-order motion. Meanwhile, although current computer vision models perform very well in estimating object motion in complex natural scenarios, these models show a different interpretation of second-order motion and generate noisy responses, as we previously observed in Figure 4.3.

In this chapter, we investigate the second-order motion problem from two aspects. First, from an engineering point of view, we ask why computer vision models can't recognize second-order motion despite coping well with complex natural scenes. We also explore how to enable our model and computer vision systems to naturally recognize second-order motion without specifically fitting second-order motion samples. From a scientific perspective, second-order motion perception in human vision has been extensively studied using carefully designed artificial stimuli (e.g., drift-balanced motion) to control first-order motion components. However, why and how the visual system has acquired this perceptual ability in natural environments remains poorly understood. What advantages or functionalities does the second-order motion perception system provide us?

With the above questions, we hypothesized that the biological system might naturally learn second-order motion perception for the purpose of estimating correct physical object motion

amidst internal optical fluctuations produced, for example, by highlights of glossy materials and refractions of transparent materials. As a proof concept, we developed a DNN-based model to process both first- and second-order motions in natural scenes. The model was based on our two-stage model proposed in Chapter 5, consisting of a trainable motion energy sensing and a recurrent self-attention network, each inspired by biological computations in V1 and MT. For preprocessing for complex second-order features, we added a second input pathway with a vanilla multi-layered convolution network. The model was trained on two distinct optical flow datasets generated by rendering random object motion: one with purely diffuse reflection (PD) and the other with non-diffuse (ND) material properties, the latter including ample optical turbulence made by specular reflections and transparent refractions. The ND-trained model demonstrated significantly better recognition of various types of second-order motion, aligning closely with human performance measured in our psychophysical experiments. Also, this performance was unachievable without the second input pathway. The results suggest that second-order motion perception might have evolved, at least partially, to help robust estimation of object motion while countering optical fluctuations under natural environments.

Ultimately, this chapter presents the first candidate that effectively generalizes both first and second-order motion phenomena in natural scenes. It aligns closely with human psychophysical responses and mammalian neurophysiological trends.

6.1 Introduction

Creating machines that perceive the world as humans do represents a significant research challenge with profound implications for both cognitive science and engineering applications. In, cognitive science, researchers endeavor to yield image-computable models of the human cognitive system, enhancing our understanding of the brain’s inner functioning.

The typical computational model in this context is the Heggerburger-like V1-MT model [152, 184], wherein V1 uses spatiotemporal filters to extract luminance-defined motion. While such prototypes effectively explain artificial stimuli like moving gratings and plaids, it is difficult to cope with more complex natural sceneries. Current Studies in the CV field, instead, focus on extracting subsequent frames’ spatial correspondence to solve the optic flow estimation. From early methods based on spatial-temporal gradients and luminance-constant assumptions [97] to current DNN-based methods [38, 171], the performance in this field has steadily improved, now capable of handling most of the established naturalistic datasets, like Sintel [18] and KITTI [113].

While the current model effectively replicates the motion from spatial shifts defined by luminance modulations (first-order motion), it does not encompass the full spectrum of the motion stimuli and all the natural scenarios, such as the flow of the transparent river, which is the particular case of the higher-order motion. Extensive studies have shown that many animals, including macaques [128], flies [172], and humans [10], can recognize not only first-order motion but also second-order motion, which is defined by higher-order image features

such as temporal and contrast modulations. A characteristic difference between first and second-order motion is a noticeable tilted pattern in the luminance spatiotemporal Fourier spectrum, which can be extracted from first-order but not second-order motion [21, 25], as illustrated in Figure 2.5. Accordingly, the Fourier-based first-order system cannot recognize the second-order motion, both direction and speed. In contrast, neurophysiological and psychophysical studies suggest that creatures able to recognize second-order motion may have different processing pathways/mechanisms independent of first-order in the visual systems [27, 83], as illustrated in Figure 6.1.

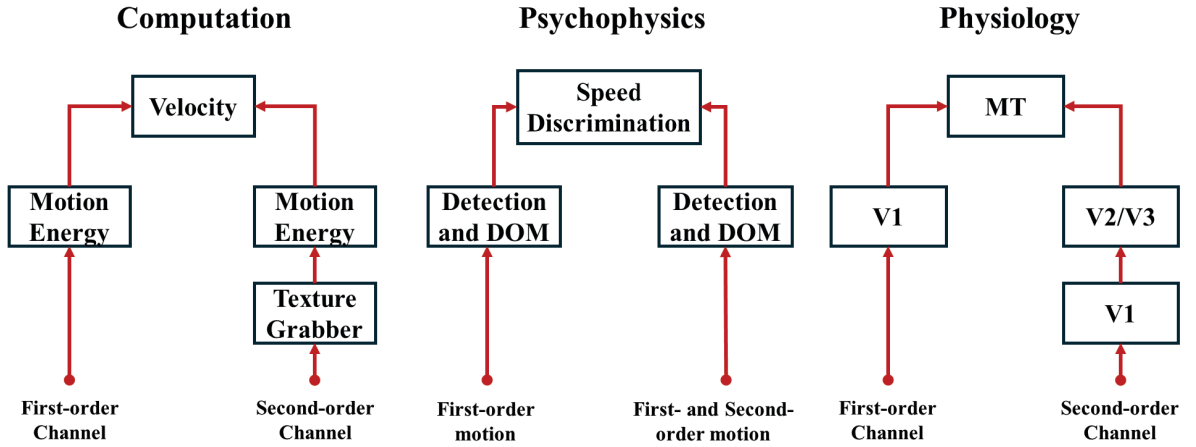


Figure 6.1: Dual channel structure in the previous study. Figure is redrawn from [27]

The wide-existing of the second-order motion system in the creatures implicitly suggests the system’s heavy need to extract second-order motion from outside the environment. However, the natural implementation of second-order motion remains poorly understood since current studies are often limited to using artificial stimuli (e.g., drift-balanced motion) to control first-order motion components. A significant challenge lies in defining specific natural scenarios to link the second-order motion systems since second-order motion signals appear relatively sparse in natural settings. For instance, state-of-the-art (SOTA) CV models, trained on “realistic” motion datasets, achieve extremely high accuracy with current natural datasets. However, they struggle to recognize second-order motion like humans (As shown in Chapter 4). This discrepancy suggests that learning to estimate natural motion does not necessarily equip AI agents with the ability to recognize second-order motion. In contrast, biological systems have independently developed this capability, evolving through dynamic interactions with natural environments. This paradox prompts questions about the meaning of second-order motion in natural settings and the potential advantages that second-order motion perception might have offered to the human vision system.

To solve the question, we contend that the second-order feature is quite ample in ‘real’ daily life rather than the video clip used to train the current model. First-order motion is

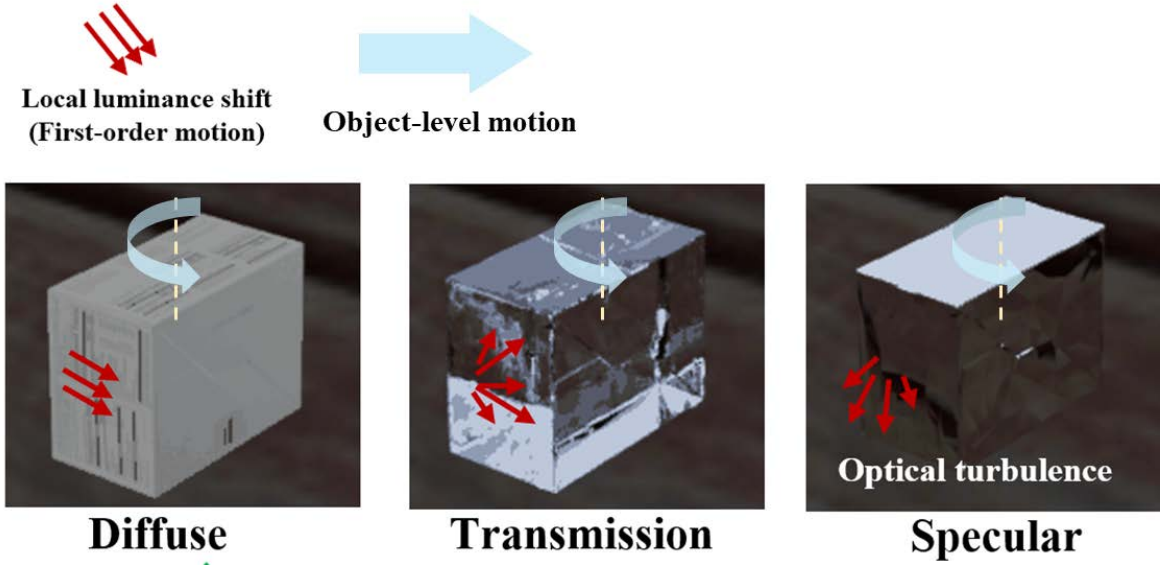


Figure 6.2: The optical turbulence on the surface with non-diffuse materials. Simulated by Blender and spatiotemporal gradient.

typically characterized as brightness shifts, adhering to a constant-brightness constraint, i.e., $I(x, y, t) = I(x + \delta_x, y + \delta_y, t + \delta_t)$, where the optic flow directly corresponds to the actual movement of objects, as shown in Figure 6.2.

However, non-first-order motion arises when local optic flow is interfered with and does not correctly indicate an object’s physical motion. For example, the background is unequally illuminated, the viewing point is unstabilized, etc. One of the most significant contributors to this discrepancy is the optical turbulence generated by objects composed of non-diffuse materials. The reflective properties of specular and glossy materials, for example, can dramatically alter an object’s original light reflectance pathways. This alteration is not merely a distortion but a dynamic, nonlinear transformation that intensifies with the object’s motion and changes in lighting conditions. Transparent materials add further complexity, creating layered optical effects that significantly impact optic flow. While theoretical models like the bidirectional scattering distribution function (BSDF) [127] attempt to encapsulate these phenomena, their predictability is compromised in real-world settings due to a complex mix of material attributes and environmental influences. We propose that these intricate, untrackable optic turbulences align closely with second-order motion cues, which are beyond the capabilities of pure first-order systems. Our hypothesis posits that the second-order visual system plays a critical role in complementing the first-order system. It is uniquely equipped to interpret and counter the optic flow turbulence and noise generated during the motion of objects, particularly those involving complex material properties. This advanced processing is essential for accurately discerning the true motion of objects amidst the chaotic optic signals prevalent in natural settings.

As a proof concept, we developed a DNN-based unified model to process both first- and

second-order motions in natural scenes. The model was based on our prototype two-stage model proposed in Chapter 5, consisting of a group of trainable motion energy sensing to capture local motion and a recurrent self-attention network for adaptive motion integration and segregation, each inspired by biological computations in V1 and MT. For preprocessing for complex second-order features, we added a second input pathway with a vanilla multi-layered 3-D convolution network and constructed a double-channel network. Such design not only aligns with the two-channel processing pathway observed in physiological studies [28,84] but also enables the model’s hardware ability to extract non-linear spatiotemporal features related to second-order motion. More importantly, we rendered a unique naturalistic optical flow dataset by manipulating the material parameters of the objects while keeping the object, illumination, scenes, and motion variables within the same distribution. This results in two distinct datasets: one characterized by purely diffuse reflection (PD) and the other featuring non-diffuse (ND) material properties; the latter includes ample optical turbulence made by specular reflections and transparent refractions. Our strategy is to train the model, especially for second-order motion extractors, to recognize the correct physical object-level motion while countering the optical interferences during the motion process. We aim to elucidate the relationship between second-order motion processing capabilities and material properties in various environments by systematically manipulating the first-order and second-order channels on the different datasets.

To precisely quantify how material properties impact second-order perception, we also contribute the first dataset of second-order motion with more naturalistic modulation. Furthermore, we enable the model to spontaneously understand the second-order motion and eventually develop a system that simulates the human motion perception system, capable of accommodating both first-order and second-order motions with certain neurophysiological and perceptual similarities to previous observations. At the intersection between computer vision models and biological computing, this study represents an effort to meld classical neuron computation in human vision science with advanced deep learning technologies. It presents the first framework that effectively generalizes both first- and second-order motion phenomena, extending from classical experimental stimuli to complex natural scenarios. While detailed results and discussions will follow in subsequent sections, it is notable that our initial findings show a promising correlation with human psychophysical responses in naturalistic first/second-order motion scenes and appear to align with key trends observed in mammalian neurophysiological studies.

6.2 Results

We begin by introducing the fundamental motion processing pipeline of our dual-channel model. Subsequently, we explore the model’s advanced motion integration capabilities, ranging from experimental stimuli in vision science to complex natural scenarios. Following this, we extend the model’s application to higher-order motion processing, powered by subordinate high-order motion channels. We will demonstrate the relationship between material properties and the

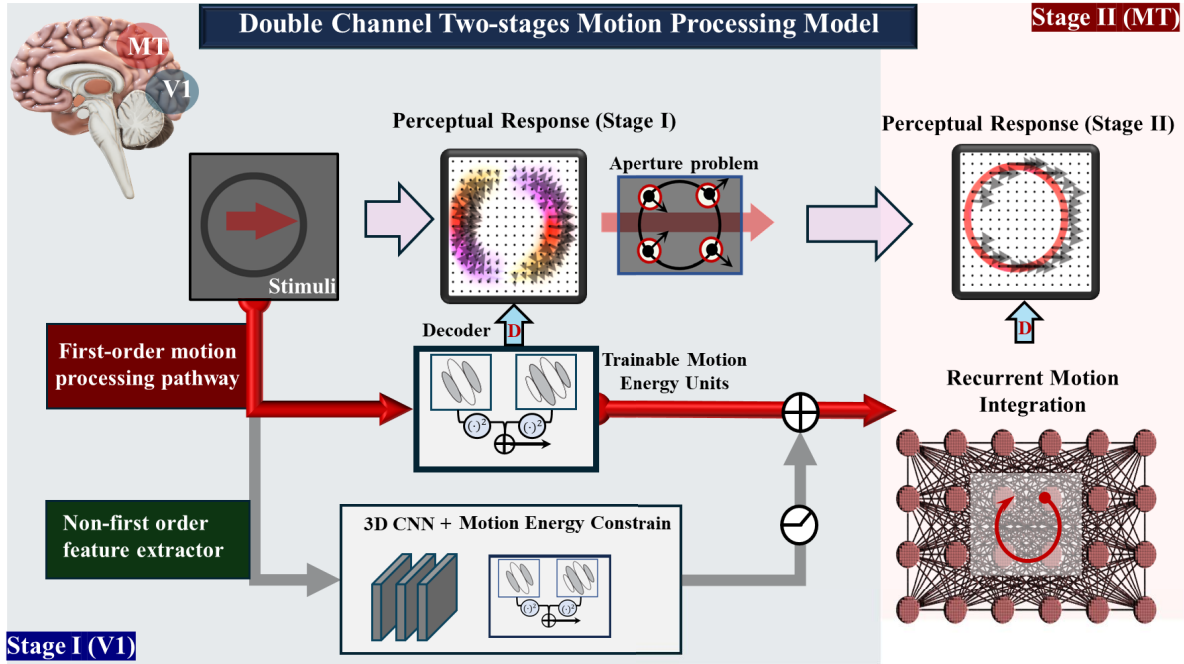


Figure 6.3: Two-Stages motion perception system prototype. The first stage mimics the V1 function by detecting local motion, while the second stage leverages a graph network for recurrent motion integration and segregation. Stage I incorporates dual channels to process both first-order and second-order motion; the first-order channel focuses on capturing Fourier-based motion using motion energy units, whereas the second-order channel employs norm 3D CNN layers to extract higher-order features.

model’s ability to perceive second-order motion. Furthermore, we employ several specialized benchmarks to quantify the role between first-order and higher-order motions and compare the performance of second-order motion against current representative models in realistic scenarios.

6.2.1 Two-stages Processing Model

Our prototype model involves two-stage motion processing that combines the classical motion energy sensors (Stage I) and modern DNNs (Stage II). Stage I is designed to capture local motion energy, simulating the function of the V1 area. In contrast, Stage II focuses on global motion integration and segregation, simulating the primary function of the MT area. The red route in Fig. 6.3 depicts the classical first-order motion process. Specifically, we built 256 trainable motion energy units, each composed of quadrature 2D Gabor spatial filters and quadrature temporal filters, to capture spatiotemporal motion energy of input videos in a multi-scale wavelet space. The key implementation difference from previous motion energy models is that we embedded computation in the deep learning framework, i.e., each motion energy neuron’s parameters, such as preferred moving speed and direction, are trainable to fit the dataset. Stage

II incorporates the modern DNNs connected with the previous motion energy model. It builds a fully connected graph on local motion energy, where each spatial location is regarded as a node, and all nodes are interconnected. We adopt a self-attention mechanism to build the graph’s topological structure, and the recurrent motion integration is a performer based on the graph to generate global motion interpretation and solve motion aperture problems (Stage II, Fig. 6.3). The whole model is supervisedly trained to estimate the pixel-wise object motion within the naturalistic datasets [18]. See the method part for details.

The motion energy channel is limited to capturing first-order motion. To enable the extraction of higher-order motion information, we introduce an alternative channel, depicted by the gray route in Fig 6.3, which uses multi-layer 3D CNNs ahead of motion energy computation to extract nonlinear spatiotemporal features. This dual-channel design draws inspiration from observations in previous studies [28].

6.2.2 Materials Properties & Second-order Motion Perception

Despite incorporating a second channel aimed at extracting non-first-order features, our model remains unable to identify second-order motion when trained solely on existing motion datasets. This limitation reflects broader challenges within the CV field, where DNN-based models struggle to spontaneously capture second-order motion perception through training on conventional motion estimation tasks [167]. To validate our initial hypothesis, we constructed a unique dataset while controlling object material properties, creating two distinct subsets: one exhibiting Lambertian reflection and the other featuring non-Lambertian properties, including glossy, transparent, metallic surfaces, etc (Fig. 6.4). The goal is to train the model, with a particular focus on second-order motion extractors, to identify physical object-level motion while mitigating optical interferences caused by non-Lambertian reflection during motion.

To quantify the second-order motion perception ability, we developed the first benchmark using natural images with second-order modulation, including classical drift-balance motion and naturally occurring phenomena such as water waves and swirl flow fields (Fig. 6.5). These scenarios, often indiscernible in Fourier space, introduce optical challenges such as chaotic reflections and refraction. The human psychophysical study indicated a strong correlation (mean=0.97, std= 0.02) in detecting various second-order modulations(Fig. 6.6), in contrast to the lower correlation (mean=0.23, std=0.13) observed in representative CV models, RAFT.

Our findings, depicted in Fig. 6.7 B, indicate that both the training set’s material properties and the model’s structural design significantly affect second-order motion perception. Training with non-Lambertian materials markedly enhanced the recognition of all second-order motion types, approaching human-level precision (mean correlation = 0.92). In contrast, the original motion energy model and computer vision model, RAFT, even trained with non-Lambertian data, showed limited ability in second-order motion perception, particularly for drift-balanced motion. In Fig. 6.7 A, we investigate the direction-tuning abilities of the first-order and higher-

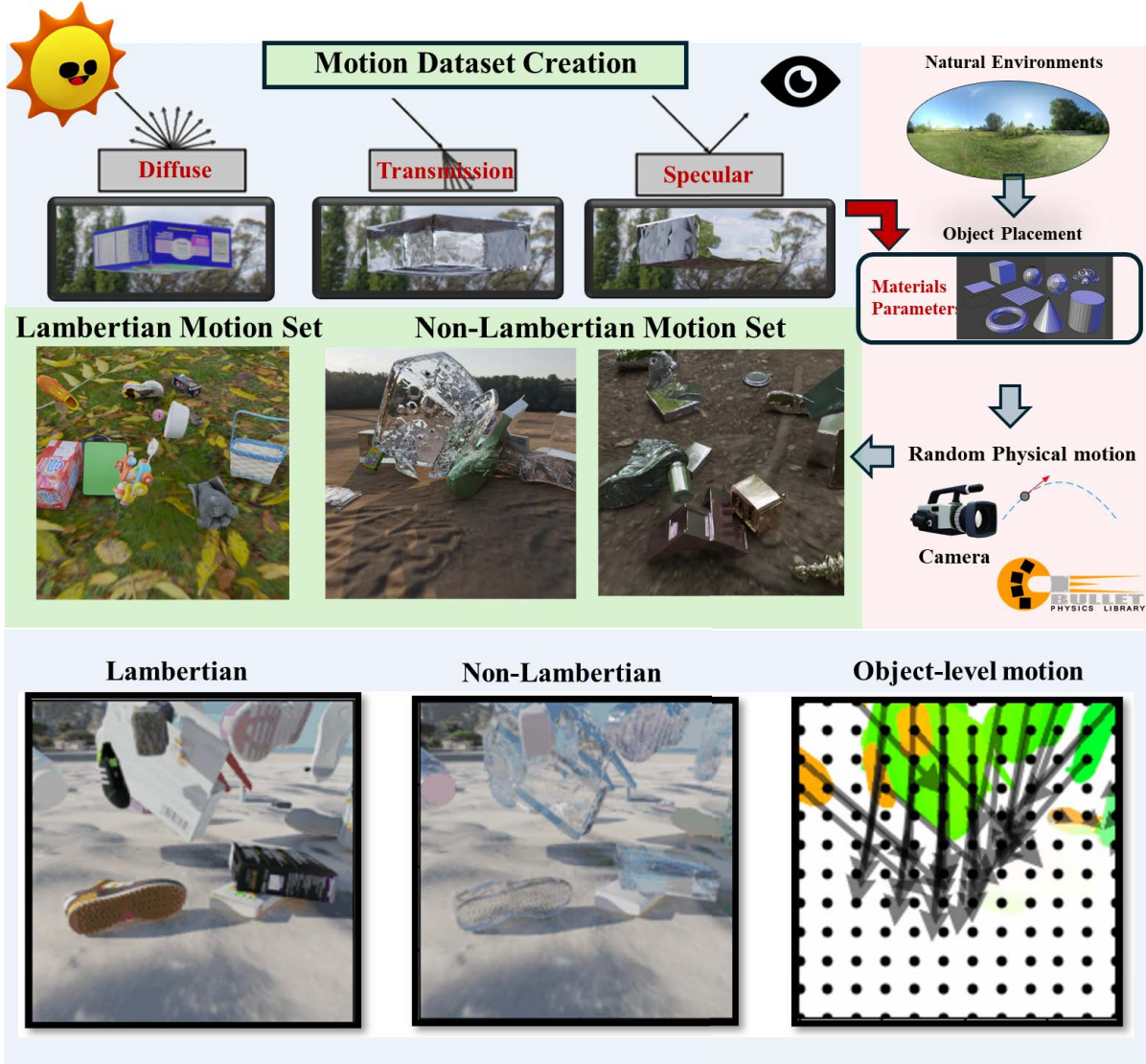


Figure 6.4: Material-controlled natural motion dataset. We manipulate material properties to create two extensive motion datasets with optical flow labels; one features purely reflective materials, while the other includes non-Lambertian materials such as specular, glossy, translucent, and anisotropic surfaces.

order motion channels using in silico-neurophysiological methods. Testing first- and second-order drifting grating in various directions and analyzing the results with a modified circular variance [106], we observed that the motion energy (ME) units primarily responded to the first-order motion. In comparison, the second-order channel exhibited enhanced tuning ability for second-order motion, further improved by training on non-Lambertian materials.

Additionally, we evaluated second-order motion perception using our benchmark, comparing the Pearson correlation between models' response and carrier motion against multiple SOTA optical flow estimation models, such as RAFT, GMFlow, and the multi-frame method VideoFlow

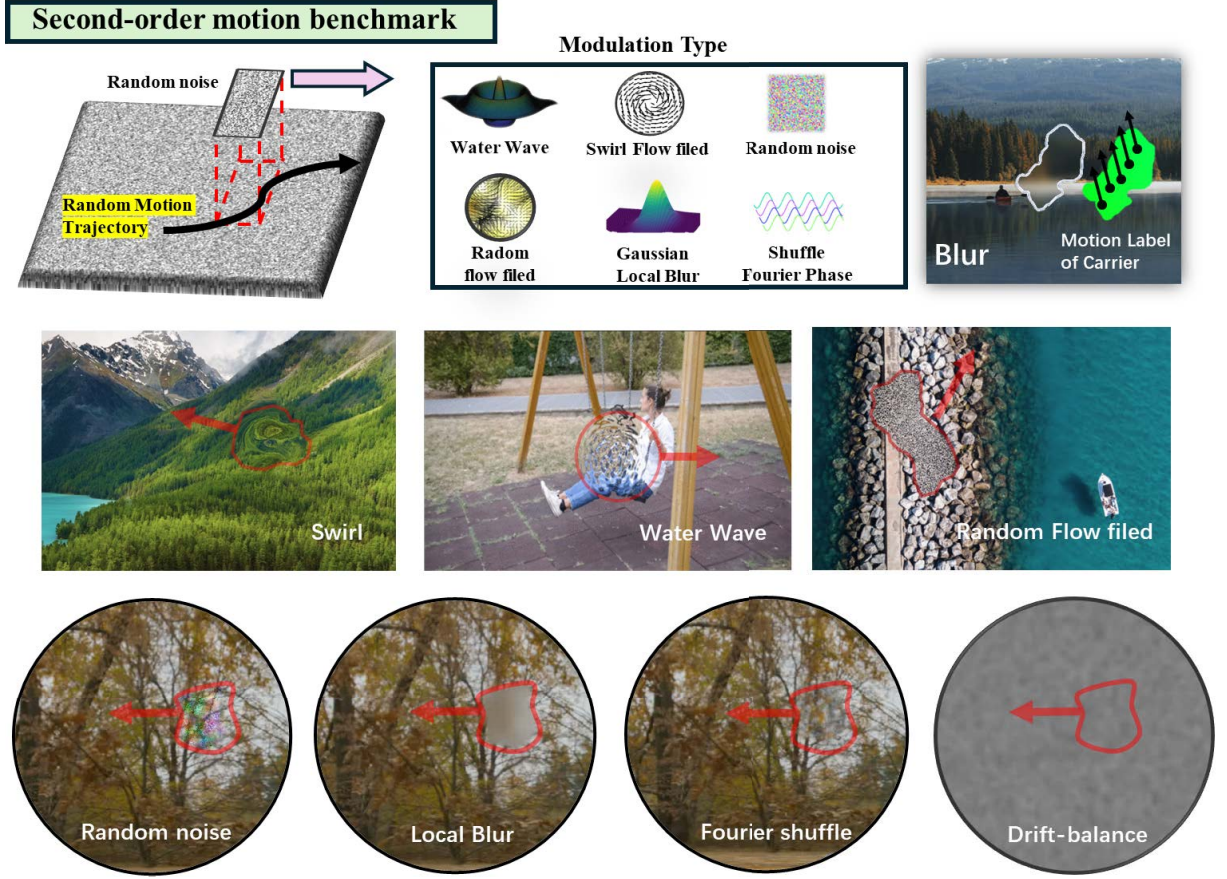


Figure 6.5: Second-order Benchmark Demonstration. The first large second-order benchmark is generated through more naturalistic modulations, such as water waves and swirl effects, applied to natural images. This dataset is utilized to either train models for second-order motion estimation tasks or to quantify their capability in recognizing second-order motion.

[151]. As shown in Fig. 6.7 (A), our model exhibited the highest correlation and stability in this context, closely approximating human performance levels. Besides, We found that while non-Lambertian properties assist second-order motion perception in most models, current CV models still struggle to recognize second-order motion, exhibiting lower correlation and high variance. This discrepancy may be attributed to structural differences, as our model is built on spatiotemporal filters that aim to emulate human motion processing mechanisms, whereas CV models primarily focus on tracking explicit pixel correspondence between frames. Second-order motions, such as drift-balanced motion, lack such correspondence across frames, leading these models to generate noise. In Figure 6.8, we further show more quantitative demonstrations of the interplay between material properties and the model's response to the second-order motion.

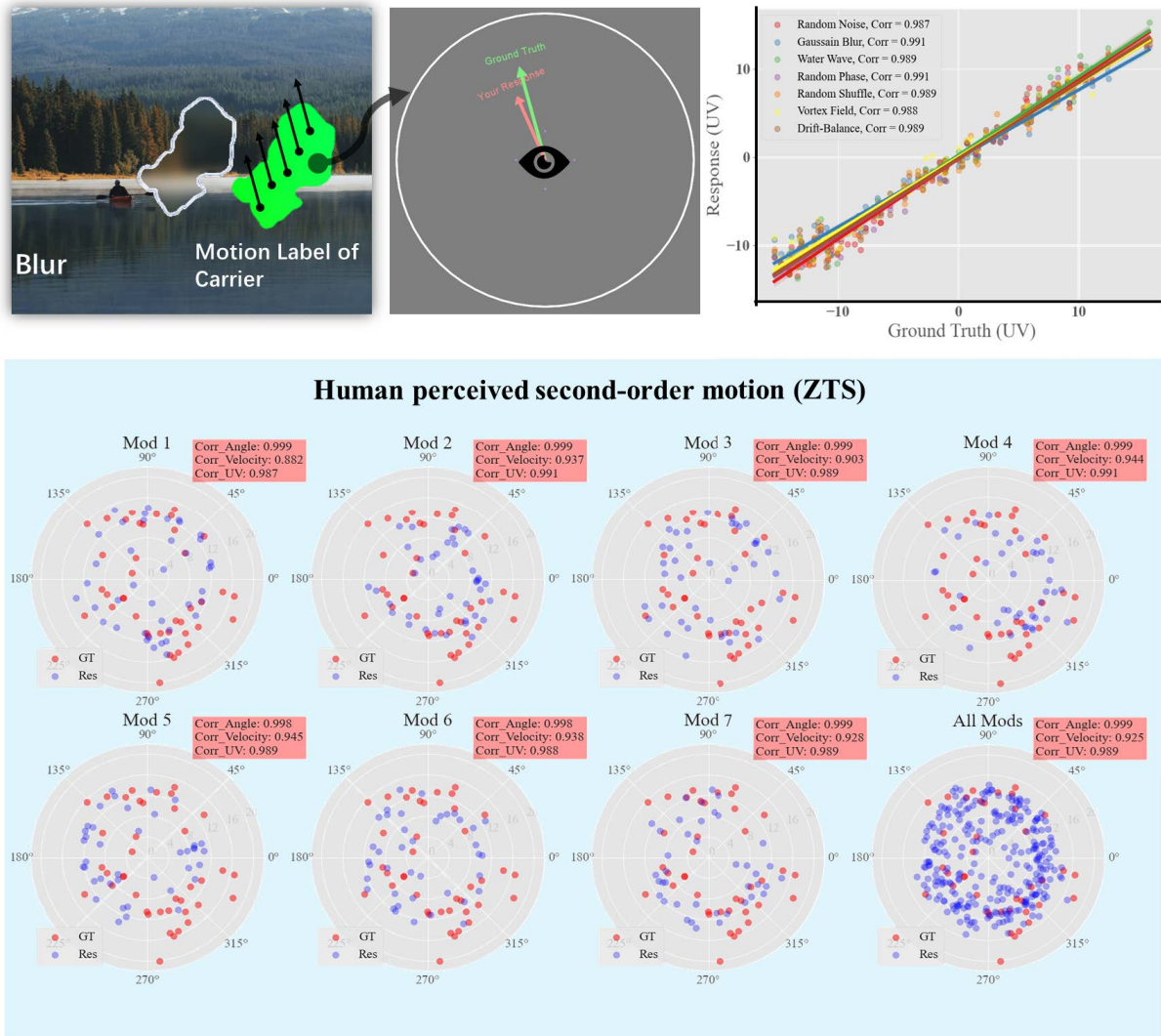


Figure 6.6: Human response on second-order motion. Our psychophysical experiments demonstrate that humans reliably and accurately perceive a range of second-order motions.

6.3 Discussion

DNNs have demonstrated remarkable capabilities across a variety of vision tasks, and their potential to explain the HVS is an active area of research. Recent studies have leveraged DNNs to probe and decode the neural and perceptual mechanisms underlying visual motion. For instance, Rideaux et al. [143, 144] and Nakamura and Gomi [118] utilized multilayer feedforward networks, while Storrs et al. [161] employed a predictive coding network (PredNet). These studies discovered notable parallels with neurophysiological data in modeling biological visual motion processing. Similarly, De Jong et al. [32] observed that certain units in FlowNet exhibit spatiotemporal frequency tuning properties akin to mammalian neurons. DorsalNet [115] adopts first-person perspective video stimuli to train a 3D ResNet model, enabling it to predict self-

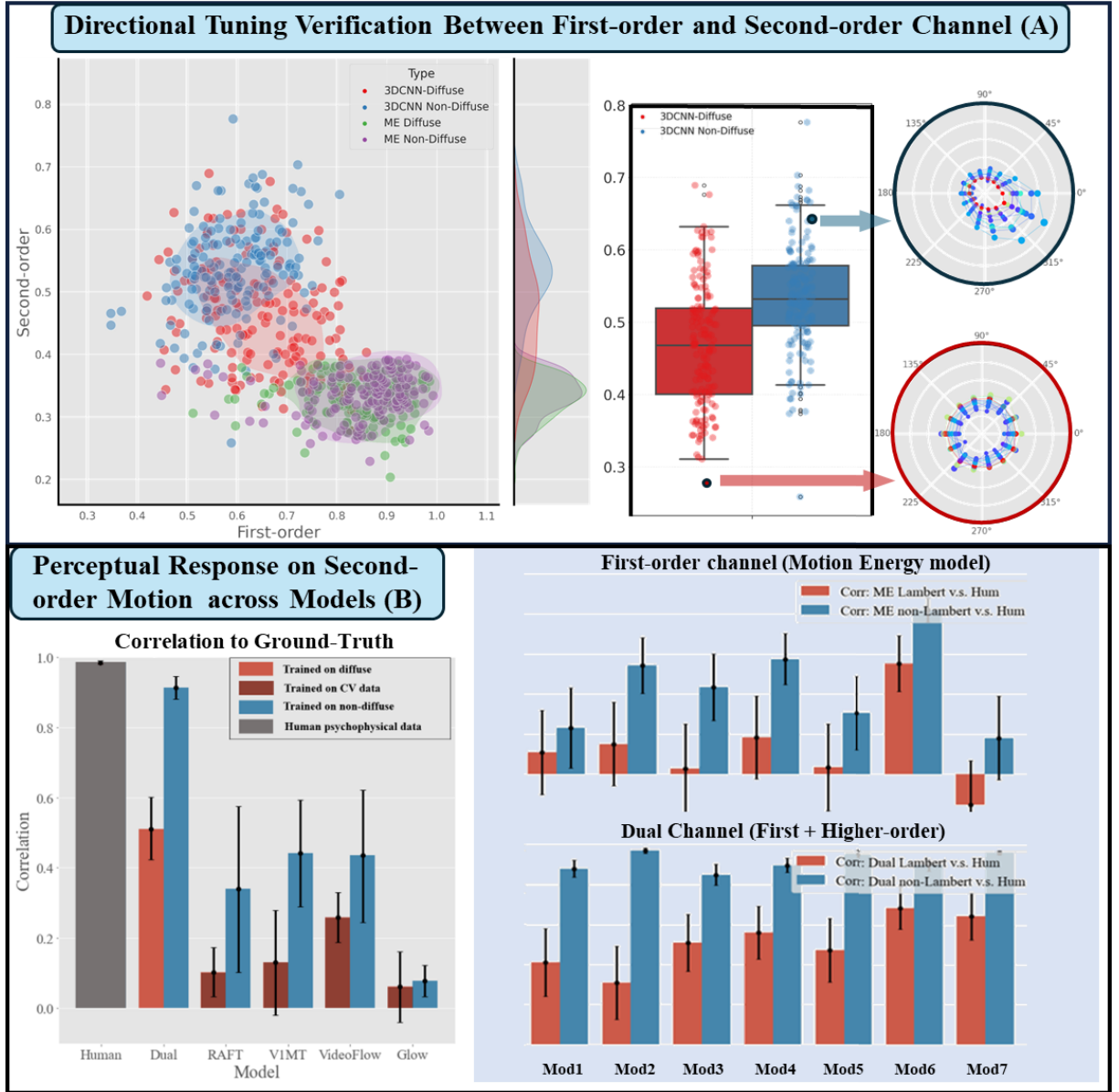


Figure 6.7: Interplay between material properties and second-order motion perception. **(A)**: Directional tuning curves of model units are tested using first-order and second-order gratings, with a modified circular variance measure quantifying directional tuning (ranging from 0 to 1, where 0 indicates a uniform distribution). **(B)**: (Left): Pearson correlation across various types of second-order motion. The error bars represent a 95% confidence interval. (Right): Detailed Pearson correlation across all modulation types. Mod1 to Mod7 represent random noise, Gaussian blur, water wave, Fourier phase shuffle, random pixel shuffle, swirl, and drift-balanced motion.

motion parameters and recapitulate the neural representation of the dorsal visual stream. With the same goal in mind, our research seeks to trade off the mechanisms of HVS and the engineering potentialities, establishing a comprehensive model capable of replicating human multi-order

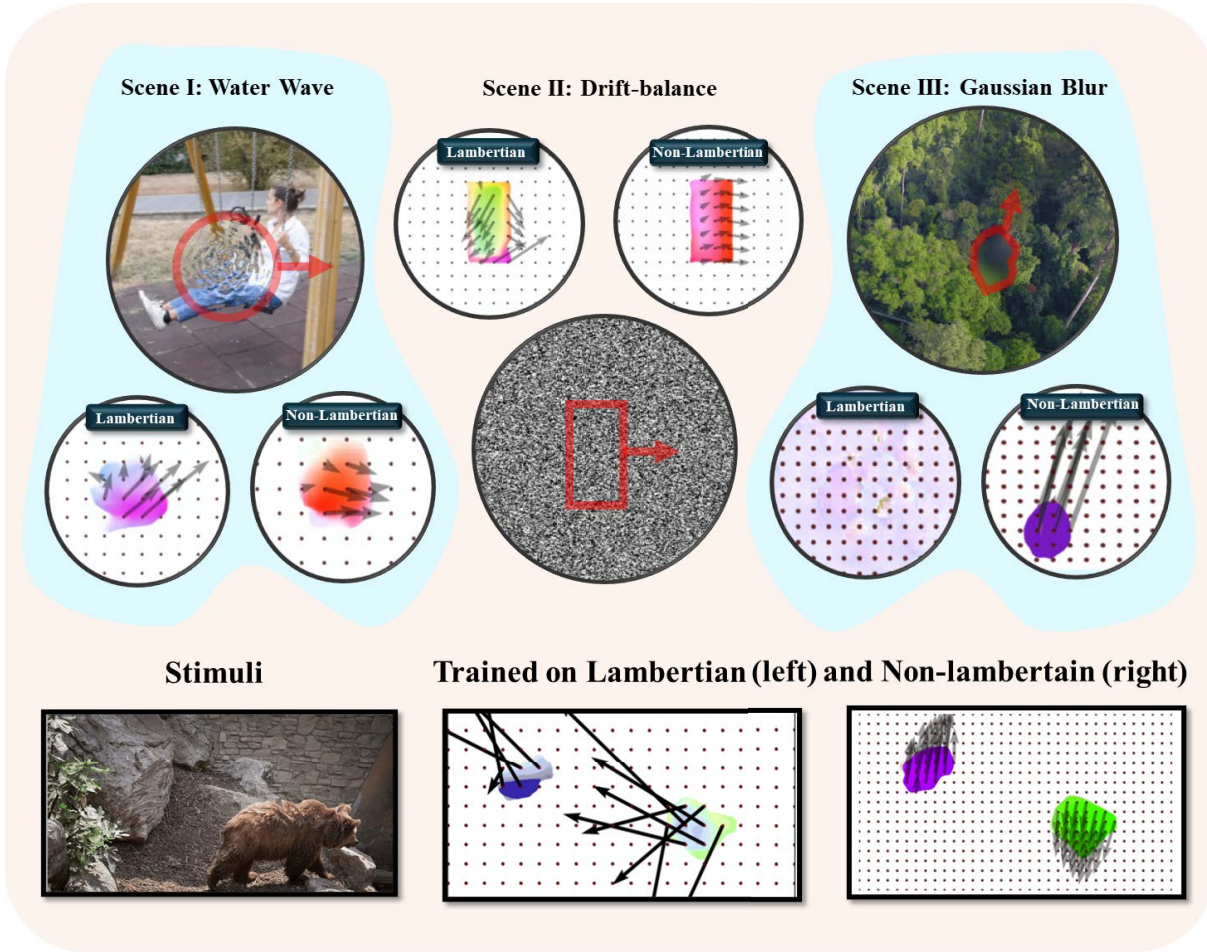


Figure 6.8: Demonstration of the interplay between material properties and second-order motion perception.

motion perception across both experimental and complex natural settings.

A generalized computational model: We simplified the biological motion process pipeline and proposed a V1-MT architecture that models the complete pathway from images, through neural representations, to the perceptual response. Through end-to-end training using a wide range of datasets, our model generalizes well from simple stimuli to complex natural scenes and partially captures important characteristics of motion-processing neurons, including a change in spatiotemporal tuning from V1 to MT areas. In addition, we integrate a high-order motion channel into our framework, enabling the model to spontaneously develop second-order motion detection capabilities at a level comparable to human performance.

To model the motion integration function, we introduced a novel recurrent process based on the attention mechanism. This process successfully explains a wide range of physiological findings (e.g., a change in the population of component and pattern cells from V1 to MT) and psychophysical findings (e.g., global motion pooling). It also improves the partial correlation with human psychophysical response. The success of the attention mechanism in motion in-

tegration could be attributed to its similarity to the human visual grouping mechanism [109] or similar feature grouping/binding that may be accomplished using a top-down attentional selection mechanism [175].

The Interplay between First-order and Higher-order Channels: We aim to comprehend the roles of first-order and second-order motion perception in nature. We propose that optic turbulence caused by non-Lambertian reflection significantly impacts second-order motion. Although some research suggests that second-order motion might be related to transparency [31, 148], material perception [10, 123, 131], or some invisible object inference [3], a systematic quantification and verification of these factors have not been investigated yet.

As a first candidate, we employ a biological-orientated model and controlled material properties to systematically verify the relationship between second-order motion and material properties. The results, shown in Fig. 6.7, clearly support our hypothesis. Further, as illustrated in Fig. 6.9, supplementary experiments were conducted to elucidate the interplay between the first-order and higher-order channels, which details an experiment that assesses the impact of the higher-order motion channel. We trained three distinct motion models to estimate realistic motion and evaluated their performance on non-Lambertian and Lambertian surfaces:

- Type I: Single-channel model (pure motion energy model).
- Type II: Dual-channel model with a higher-order channel pre-trained for second-order motion estimation, with fixed parameters during joint training.
- Type III: Fully trained dual-channel model.

Performance evaluation using end-point error (EPE: lower is better) on both Lambertian and non-Lambertian datasets showed significant enhancement in performance on non-Lambertian datasets when integrating a fixed second-order extractor (Type II) into the first-order channel.

We also highlight the second-order channel’s effectiveness in motion estimation under extreme noise conditions in natural scenarios, such as turbulent water flow and moving box with water fluctuation (Fig. 6.10). We humans can perceive the local optical noise generated by fluctuating water. However, globally, we can also infer the general moving direction of the flow. This is highly related to the functionality of second-order motion. The isolated second-order motion may be rarely observed in natural environments; more often, it coexists with first-order motion as the optical fluctuation. The computer vision models failed to consider this aspect, as they are largely affected by local optical noise and generate broken optical flow fields. Hence, we hold the view that joint consideration of first- and second-order motion may potentially make the vision system more stable and reliable even in complex natural environments.

First-order motion, characterized by luminance shifts under a constant-brightness constraint, correlates directly with physical object movement, a principle widely utilized in CV fields such as the Lucas-Kanade methods. However, non-first-order motion occurs when local luminance shifts are influenced by optical turbulence from non-Lambertian materials, failing to indicate

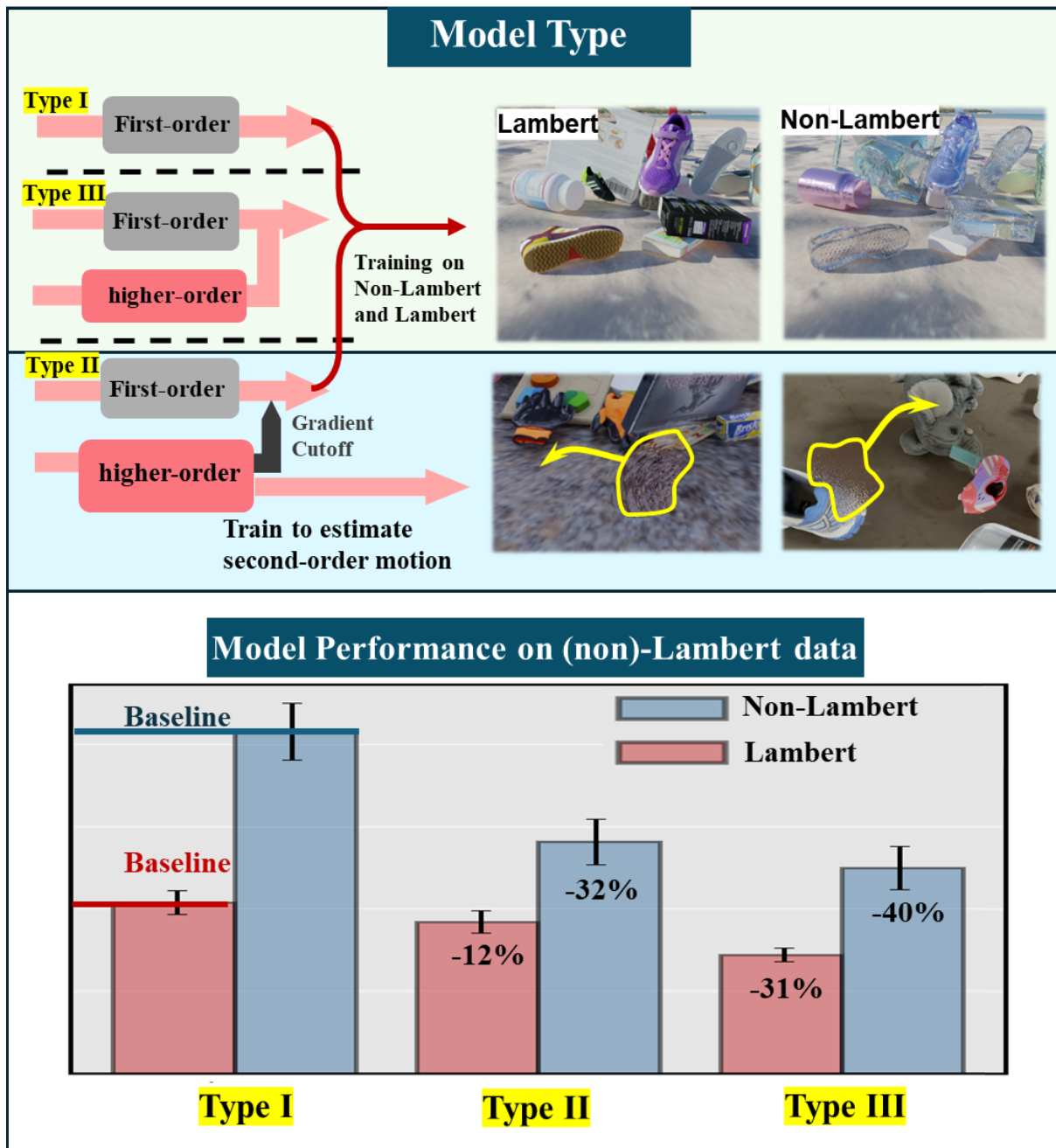


Figure 6.9: The roles between first-order and higher-order motion channels: The efficiency of the higher-order motion extractor is evaluated using both Lambertian and non-Lambertian datasets. We explore three model configurations: an ME model restricted to first-order motion extraction, an ME model with a frozen second-order motion extractor (ME+SEC), and an ME model with a trainable second-order channel. Findings suggest that the primary first-order motion system is capable of non-Lambertian motion estimation, while the second-order channel enhances the primary system’s ability to counteract material optical noise, yielding more stable results.

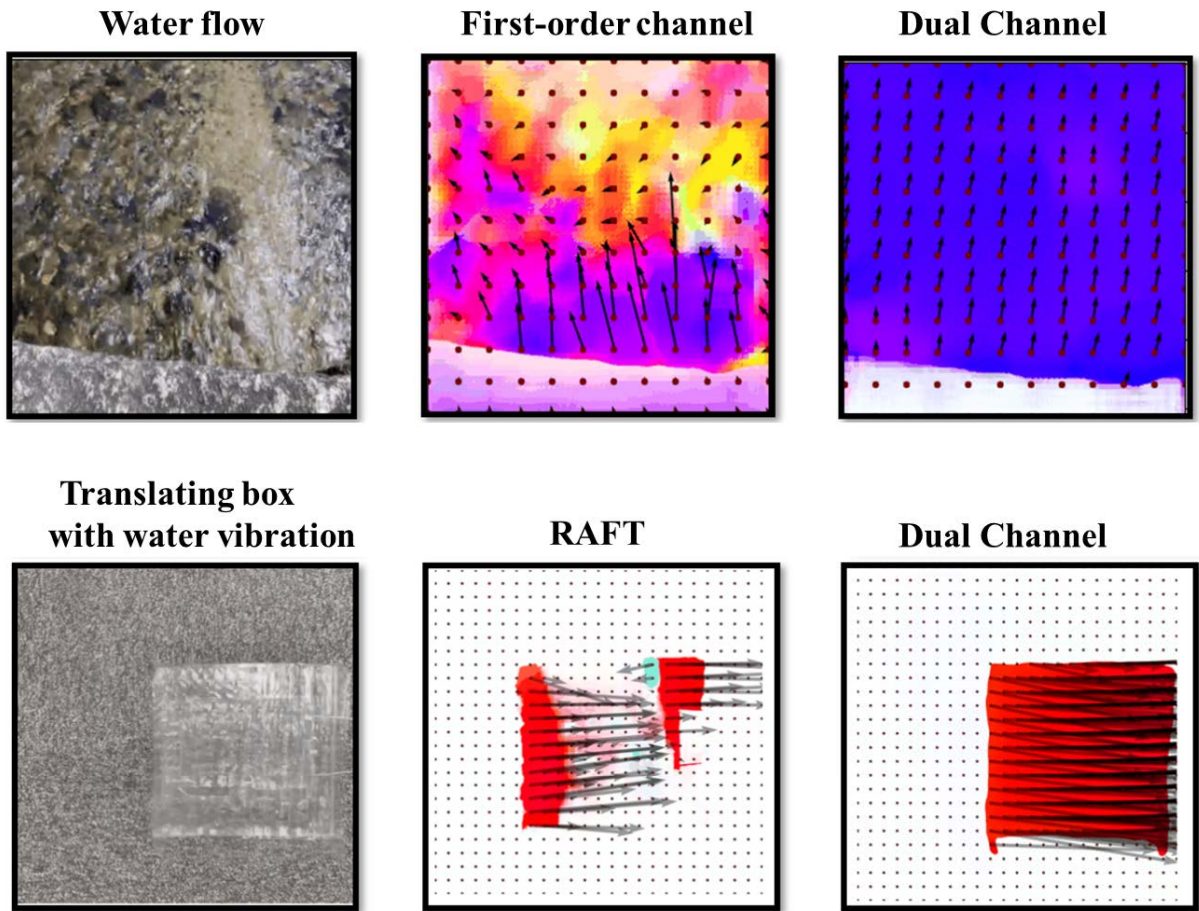


Figure 6.10: Second-order motion in natural environment. Up: water flowing in the river; Down: the moving box with the vibration water. We humans can perceive the local optical noise generated by fluctuating water. However, globally, we can also infer the general moving direction of the flow.

the correct physical motion. The reflective properties of specular and glossy materials can significantly alter light reflectance pathways, casting noisy-like optical interference, commonly observed in natural settings with liquid fluctuations and extreme weather conditions. In such complex optical environments, a higher-order motion detection system is necessary to aid the primary vision system in accurately tracking object motion. Combining the results in Fig. 6.7 and Fig 6.9, it is suggested that second-order motion perception has evolved, at least partly, to ensure reliable object motion detection amid optical turbulence in natural settings.

Overall, the results underscore that human motion perception is not solely dependent on “motion” per se. It involves an intrinsic interaction with higher-level vision functions beyond brightness shifts. The human vision system has evolved specialized channels and mechanisms to handle this complex information, incorporating separate first-order and higher-order motion processing areas [27, 83], along with the ventral and dorsal streams, for distinct functionalities.

Insights from such biological concepts have influenced numerous computer vision works, such as the Two-Stream Net [153]. Similarly, our dual-channel strategy demonstrates promising results in extending basic first-order motion perception to include higher-order motion perception, which is closely related to other vision functions like depth estimation, layering, and segmentation. It is anticipated that a foundational vision model that links multiple vision functions will continue to be a progressive direction for better understanding and simulating the human vision system.

Limitations: Our study highlights the inadequacy of motion energy-based models in capturing higher-order features in specific scenarios, such as non-Fourier motion depicted in Fig. 6.5. To address the limitation above, human-like vision systems require higher-order feature extractors beyond basic first-order motion systems. While our approach integrates multi-layer 3D CNNs to serve this role, it inevitably compromises the model’s transparency and explainability. We can hardly interpret which high-order features that being extracted and utilized to infer the optical flow, such as long-temporal statistics, depth cues, occlusion, structural constraints, or layer estimation. Actually, these features surpass mere motion perception and touch the integration of the broad function of the human vision system [123].

Additionally, we employ a simplistic, linear fusion approach in our model to merge first-order and higher-order channels, while the interaction between first- and second-order signals remains controversial in the current study, and may be much more complicated for the real human system [100]. Considering these challenges, future research may focus on developing more controllable feature extraction methods, which would clarify the roles of specific features or cues in relation to particular scenes. Moreover, first-order and higher-order features each have their merits in different scenarios, and an adaptive combination and interaction offer a promising avenue for further exploration.

Impacts on Computer Vision Models: our study suggests the potential of drawing inspiration from the HVS to mitigate the limitations and instabilities inherent in contemporary CV models. For instance, the instability of CV methods in certain scenarios is often linked to their focus on matching pixel correspondence between frame pairs [141]. This strategy differs from the human motion perception mechanism, which depends on dynamic spatiotemporal features and demonstrates exceptional stability and adaptability in interpreting physical object motion. As we observed, the integration of a second-order motion system, adept at extracting long-term, higher-order features, is essential in aiding the primary system to estimate object motion accurately. This addition not only combats noise and optical turbulence but also results in a more stable and reliable motion estimation model, particularly useful in challenging scenarios such as adversarial attacks [141] or extreme weather conditions [147]. We believe these advancements offer significant insights to enhance the motion estimation in CV field to develop a more reliable and stable model.

6.4 Methods

6.4.1 Model Structure

We explain our biological-oriented model into two stages: Stage I and Stage II. As illustrated in Fig. 6.3, Stage I comprises two channels: the first channel simulates the initial biological system with motion energy constraints, while the second channel uses 3D CNNs to enable non-first-order feature extraction.

The details of the first-order motion channel structure are described in Chapter 5. In the second-order channel, we employ standard 3D CNNs to extract non-first-order features. This channel consists of five layers of 3D CNNs, each with a kernel size of $3 \times 3 \times 5$, interconnected through residual connections and non-linear ReLU activation functions. The 3D CNN layers serve as a preprocessing step for extracting non-linear features, which are then processed through motion energy constraints, as previously described, to calculate the motion energy. The input to this channel is a sequence of RGB images, and the output is formatted to match the first-order channel, denoted as $\mathbf{E}_2 \in \mathbb{R}^{\frac{H}{8} \times \frac{W}{8} \times 256}$. Both the first-order and second-order channel activations undergo the same normalization process, after which they are merged using a 1×1 convolution. This fused output is then fed into Stage II.

6.4.2 Training Strategy

In this work, we use a supervised learning approach, guided by the similarity between human motion perception and physical ground truth (GT) as suggested by [189]. Our focus is on how well the model mimics human motion perception, rather than its precision in predicting GT. Our dataset includes various natural and artificial motion scenes, incorporating the Sintel benchmark [18], and our custom multi-frame datasets with non-textured and drifting grating motions. This diverse dataset helps the model adapt to different non-textured scenarios and introduces a slow-world Bayesian prior [184].

For studying second-order motion, we develop datasets for Lambertian, non-Lambertian, and second-order motions, integrating them into our training to handle specific verifications. Throughout the training, we use a sequential pixel-wise mean-square-error loss to minimize the difference between GT and model predictions.

Environment: We conducted model training using the PyTorch 2.0 framework on a workstation with four Nvidia RTX A6000 GPUs in parallel, utilizing CUDA 11.7 runtime. Data analysis was carried out using MATLAB and Python libraries, including SciPy and Pandas.

Timing Setting: Accounting for the standard playback frame rate of 25 FPS and the human visual impulse response duration (approximately 200 ms), we configured the temporal window of Stage I to cover 6 frames (200 ms). For training, sequences of 11 consecutive images (400 ms) are inputted into the model. Supervised training utilizes the instantaneous velocity at the sequence’s midpoint, specifically the fifth frame, as the training label. The second-order

channel, employing 3D-CNN, is designed with an extended temporal sequence of 15 frames (600 ms) to effectively capture long-term spatiotemporal features.

6.4.3 Datasets Generation

Dataset Rendering

We utilized Kubric’s pipeline [52] to generate extensive motion datasets, integrating Py-bullet [30] for physics simulation and Py-blender [14] for rendering. Our approach involved selecting a variety of 3D models and textures from resources like ShapeNet and GSO, coupled with natural scene HDRI backgrounds from Polyhaven [196] for realistic lighting and textures.

In the process, objects were positioned at specific heights in the 3D scene. Using Py-bullet, we simulated the physical dropping and interactions between objects. Camera placements were carefully chosen for optimal viewing angles. Blender then rendered these 3D scenes into 2D images, simultaneously generating a detailed motion flow field for each pixel, as shown in Fig. 6.5

Material properties were manipulated using the `principled` BSDF function to create natural-looking optical effects. This included generating materials with both Lambertian and non-Lambertian reflections, controlled by parameters like metallicity, specular, anisotropy, and transmission. Other aspects, such as optical flow distribution, illumination, and object and scene arrangement, were standardized to maintain consistency across the dataset.

Second-order dataset modulation

We generated two types of second-order motion datasets, named SEC-A and SEC-B. SEC-A serves as a benchmark for quantifying second-order motion perception in humans and models. It comprises 40 scenes, each with seven different types of second-order motion modulations. To fix first-order motion interference, we selected natural static images and generated random carrier motions using a Markov chain. We assume the random carrier movement $\mathbf{S}(t)$ can be decomposed into two orthogonal vectors $\mathbf{S}(t) = [U(t), V(t)]$. At any time step t , the motion state depends solely on the previous time step $t - 1$:

$$\Pr[\mathbf{S}(t)] = \Pr[\mathbf{S}(t) = s_t \mid \mathbf{S}(t-1) = s_{t-1}] \quad (6.1)$$

The motion states $[U, V]$ at time t are sampled from 2D Gaussian distributions, conditioned on the previous motion state. Each carrier is modulated by specific second-order motion modulators, including pure spatial (Local blur), pure temporal (Drift-balanced motion), and spatiotemporal modulations (Others). Random noise and Local blur involve applying sparse Gaussian noise or Gaussian blur to carrier regions. Water wave, swirl, and random flow field modulations warp local pixels with specific flow fields. For instance, for $(x, y) \in (-\pi, \pi)$ and $t > 0$, the flow field of the water wave $f_{u,v,t} = \left[\frac{\partial K}{\partial x}, \frac{\partial K}{\partial y} \right]$ is generated by simulating spatiotemporal water wave

dynamics:

$$\begin{cases} K(r, t) = \cos(2\pi fr) \cdot e^{-\gamma r^2} \cdot \cos(2\pi \xi t) \cdot e^{-\delta t^2} \\ r = \sqrt{x^2 + y^2} \end{cases} \quad (6.2)$$

Parameters f , ξ , and δ control the dynamics of the wave. By superimposing multiple water wave fields with varied dynamics on different locations, we create chaotic local optical turbulence concurrent with the carrier motion, resulting in the real carrier motion being obscured by local optical noise and invisible in Fourier space. This phenomenon epitomizes the characteristics of second-order motion. Similarly, Random flow field or shuffle Fourier phase modulation involves warping pixels or the Fourier phase of original local regions using a randomly sampled Gaussian flow field.

SEC-B is designed to train networks to directly recognize second-order motion. It includes 100 scenes, each containing 24 continuous frames. Similar second-order modulations are applied to non-Lambertian motion scenes to negate the effects of second-order factors caused by materials. The camera is also fixed to eliminate interference from global camera motion. This dataset is used specifically in the verification stage to train the second-order channel to recognize second-order motion, as shown in Fig.6.7 (C).

6.4.4 Experimental Details

We employed drifting Gabor or plaid (composed of two Gabor components) with a single frequency component as the input stimulus. For second-order motion, drift-balanced motion modulation was applied to the same Gabor carrier.

Directional Tuning: We employed a single frequency drifting-Gabor and plaid (superimposed at ± 30 degrees) as stimuli. Initially, 12 directions were uniformly sampled from $(0, 2\pi]$. For each direction, we logarithmically sampled $8 \times 8 = 64$ sets of spatiotemporal frequency combinations and obtained 64 directional tuning curves for each unit using the drifting-Gabor stimulus. The spatiotemporal frequency with the largest s.t.d. was selected as the preferred frequency st^* for each unit. Subsequently, Gabor and plaid stimuli with the frequency configuration of st^* were input into the model to derive each unit's directional tuning curve.

In Fig. 6.7 (B), we quantify orientation selectivity O_{ori} using the modified circular variance [106]:

$$O_{\text{ori}} = \left| \frac{\sum_i A(\theta_i) \exp(2i\theta_i)}{\sum_i A(\theta_i)} \right|, \quad (6.3)$$

where $A(\theta_i)$ represents the normalized response at angle θ_i .

Human experiment

Participants and Apparatus: Stimuli were displayed on a VIEWPixx /3D LCD monitor (VPixx Technologies) with a resolution of 1920×1080 pixels at a 30 Hz refresh rate. Display luminance levels were linearly calibrated, with the minimum, mean, and maximum values of

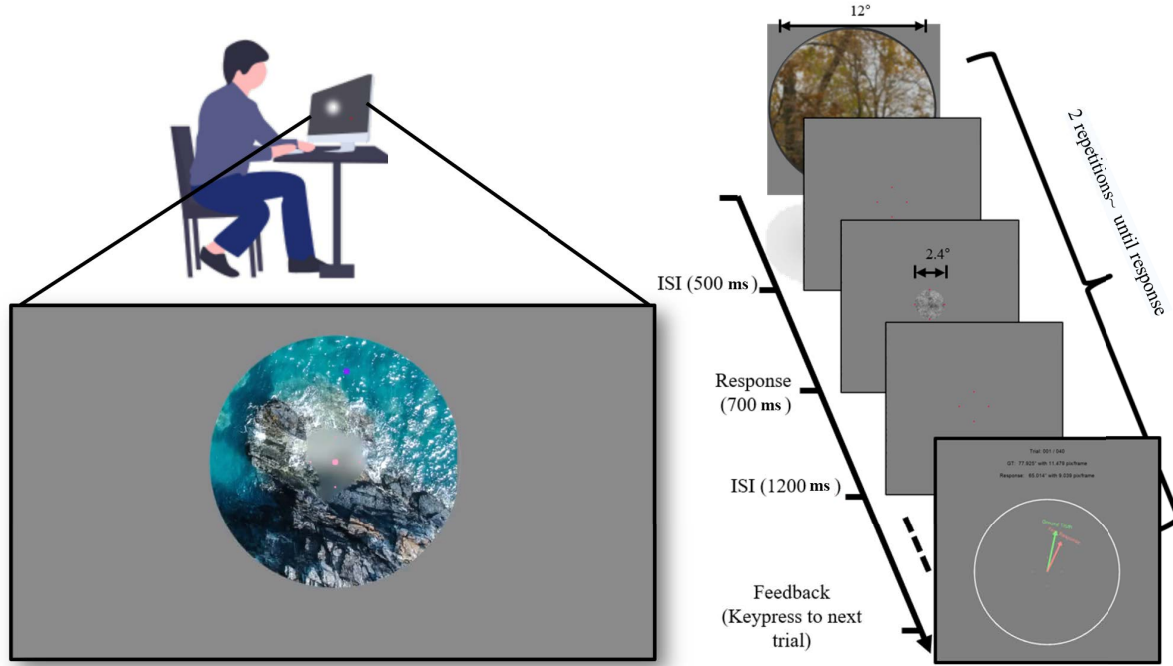


Figure 6.11: The pipeline of the human psychophysical experiment. Human subjects need to use the mouse to control the moving noise to match the second-order motion they perceive. Three subjects participated in the experiment, and we collected 280 human data for each subject.

1.8, 48.4, and 96.7 cd/m^2 , respectively, using an i1Pro chromometer (VPixx Technologies). The viewing distance was set at 70 cm, with each pixel subtending 1.2376 arcmin.

Participants in a darkened room using a chinrest for head stabilization were involved in experiments conducted using the Psychopy library [132]. The study adhered to the ethical standards of the Declaration of Helsinki (excluding preregistration) and was approved by the Ethics Committee of Kyoto University (KUIS-EAR-2020-003). Two authors and one naive participant (three males, average age 25.3) with normal or corrected-to-normal vision participated. Informed consent was obtained prior to the experiment, with participants compensated afterward.

Stimuli and Procedure: As illustrated in Figure 6.11, the protocol evaluated participants' ability to estimate second-order motion. Each trial displayed target movement within apertures (600 pixels in diameter) at the center of the screen for 500 ms (15 frames), followed by brown noise movement (120 pixels in diameter) of the same duration with interstimulus intervals of 750 ms in between. A probe (15 pixels in diameter) was used to mark the location and timing of the target motion during its presentation for estimation. Additionally, four dots (five pixels each) served as position markers, displayed in an orthogonal arrangement 60 pixels away from the display's center. Participants adjusted the direction and speed of the noise movement using a mouse to match the target's labeled motion. Seven types of second-order motion modulations were tested, each comprising 40 scenes. Four variations were presented to counter directional

bias: Original, Horizontal Flip, Vertical Flip, and Both. A total of 1120 trials per participant were conducted, taking 6 hours, with results averaged across flips to yield 280 perceived motion vectors for comparison with CV models. The stimuli sequence was fully randomized for each participant.

Chapter 7

Conclusion

7.1 Overview

This thesis examines the exploration of visual motion estimation at the intersection between computer vision (CV) and vision science (VS), with the ultimate goal of minimizing the gap between CV and the human visual system in the visual motion perception field.

We take a comprehensive journey through motion perception, from unsupervised learning of optical flow estimation in the computer vision field to human visual motion processing and perception. Ultimately, we construct a generalized motion-processing model that merges deep learning intelligence with insights from VS, balancing the functional similarity and neurophysiological plausibility of the HVS with engineering applicability to complex natural scenarios.

- The initial Chapter 1 provides a comprehensive overview of human visual motion perception and optical flow estimation, emphasizing their significance across various domains in CV and VS. This chapter also explores the historical context of research in this area, highlighting progress made and identifying persistent gaps between the two fields.
- Chapter 2 provides preliminaries associated with visual motion perception and optical flow estimation tasks. Specifically, it describes the current methodology of supervised and unsupervised learning for optical flow and identifies potential problems. Concurrently, it introduces basic computational methods for motion perception in both CV fields—such as the Lucas-Kanade method, DNN-based methods—and in VS fields, which include models based on Motion From Fourier Component (MFFC) principles (Fig. 2.3) and higher-order motion perception such as second-order motion perception.
- Chapter 3 delves into the motion perception of computer vision-based methods. Specifically, it explores the unsupervised learning method based on deep learning strategies, which has the potential to generate learning behavior similar to that of animals learning motion perception by dynamically interacting with the world. This chapter focuses on modeling unsupervised learning optical flow from multi-frame sequences of dynamic

scenes. A novel temporal dynamic modeling strategy is introduced in unsupervised learning, leveraging temporal continuity across frames to enable the model to learn motion rules with content variation and occluded regions across frames. Coupled with a well-designed adaptive fusion model, we achieve state-of-the-art performance on the Sintel Clean and Final benchmarks (Table: 3.10).

- Chapter 4 demonstrates the differences between CV models and human perception in motion perception tasks, from their inner mechanisms to their perceptual responses. We incorporated several representative CV models, including supervised and unsupervised methods, and compared them with human perceptual responses based on the in-silico psychophysical method. The results indicated that most DNN-based models do not generalize to simple experimental stimuli and cannot account for perceptual motion reversals perceived by humans with contrast-reversed patterns and a missing fundamental grating (Fig. 4.3). Additionally, none of the tested models can explain occlusion-aware motion integration and second-order motion (Fig. 4.6). Generally, there is a significant gap between human and CV models that needs to be bridged.
- Chapter 5 proposes an image-computable model of human motion perception by bridging the gap between biological and CV models. Specifically, it introduces a novel two-stage approach that combines trainable motion energy sensing with a recurrent self-attention network for adaptive motion integration and segregation. This model architecture aims to capture the computations in V1-MT while providing the ability to derive informative motion flow for a wide range of stimuli, including complex natural scenes. In silico neurophysiology reveals that our model's unit responses are similar to mammalian neural recordings regarding motion pooling and speed tuning. The proposed model can also replicate human responses to a range of stimuli examined in past psychophysical studies. Experimental results on the Sintel benchmark demonstrate that our model predicts human responses better than the ground truth, whereas state-of-the-art computer vision models show the opposite (Table. 5.1). The unsupervised learning method proposed in Chapter 3 further allows our model to spontaneously learn motion estimation by watching unlabeled videos, although its performance is weaker than the supervised method (Fig. 5.14).
- Chapter 6 discusses the issue of second-order motion and its significance in natural environments. It proposes a DNN-based model to process both first- and second-order motions in natural scenes. By creating initial second-order motion benchmarks to quantify this capability and training the model using specific optical flow datasets with varying material properties, we observed that second-order motion perception is significantly related to the material properties of training environments. This suggests that the second-order motion system might have evolved, at least partially, to aid in the robust estimation of object motion while countering optical turbulence in natural environments. The results also

demonstrated that current computer vision models fail to consider this aspect, as they are largely affected by local optical noise and generate broken optical flow fields (Fig. 6.10). Hence, the joint consideration of first- and second-order motion may potentially make vision systems more stable and reliable even in complex natural environments. Finally, we developed an image-computable model that simulates human motion perception processing, capable of accommodating both first-order and second-order motions with certain neurophysiological and perceptual similarities to previous observations.

Overall, at the intersection of human vision science and computer vision, this thesis contributes innovative solutions to both research fields. On the one hand, we achieve state-of-the-art performance in unsupervised learning, creating solutions that make computer vision models behave more like humans and predict human perception. The chapter on second-order motion provides insights into drawing inspiration from the HVS to mitigate the limitations and instabilities inherent in contemporary CV models. For instance, the instability of CV methods in certain scenarios is often linked to their focus on matching pixel correspondence between frame pairs. This strategy differs from the human motion perception mechanism, which relies on dynamic spatiotemporal features and demonstrates exceptional stability and adaptability in interpreting physical object motion. As observed, integrating a second-order motion system, adept at extracting long-term, higher-order features, is essential in aiding the primary system to estimate object motion accurately. This addition not only combats noise and optical turbulence but also results in a more stable and reliable motion estimation model, particularly useful in challenging scenarios such as adversarial attacks or extreme weather conditions. We believe these advancements offer significant insights to enhance motion estimation in the CV field, developing more reliable and stable models.

On the other hand, we also contribute to human vision science by identifying several similarities and dissimilarities in motion estimation between CV and VS. We build the first generalized model that incorporates previous knowledge in vision science and proposes a recurrent integration process based on a self-attention mechanism, successfully capturing the functionality of MT. Our model captures human perceptual responses and animal neuron patterns, making it a promising biological agent in vision science studies. Using this agent, we also achieve success in second-order motion studies, gaining a scientific understanding of its meaning and functionality in natural environments. This modeling method shows promise for understanding mechanisms and functionalities in humans and animals.

7.2 Limitations and Future studies

Deep learning techniques have demonstrated remarkable capabilities across a variety of vision tasks. These techniques not only enhance vision applications in real industrial scenarios, such as optical flow estimation in video editing and super-resolution [179], behavioral understanding in

CCTV [153], and scene perception in autonomous driving [55], but they also hold the potential to explain the human vision system. Recent studies have leveraged DNNs to probe and decode the neural and perceptual mechanisms underlying visual motion. For instance, Rideaux et al. [143,144] and Nakamura and Gomi [118] utilized multilayer feedforward networks, while Storrs et al. [161] employed a predictive coding network (PredNet). These studies discovered notable parallels with neurophysiological data in modeling biological visual motion processing. Similarly, De Jong et al. [32] observed that certain units in FlowNet exhibit spatiotemporal frequency tuning properties akin to mammalian neurons. DorsalNet [115] adopts first-person perspective video stimuli to train a 3D ResNet model, enabling it to predict self-motion parameters and recapitulate the neural representation of the dorsal visual stream.

With the same goal in mind, our research seeks to balance the mechanisms of the human visual system (HVS) and the engineering potentialities of deep learning. We first investigate the unsupervised learning method in the computer vision field, considering that unsupervised learning enables the model to capture motion rules by observing natural environments in a manner akin to animal behavior. We design a temporal dynamic scene augmentation to allow the model to grasp the rules of occlusion and content variation during motion, achieving state-of-the-art performance in computer vision benchmarks.

However, in Chapter 4, we identify a significant gap between current computer vision models and human perception. This gap arises from multiple factors, including model architecture, training methods, and training data. To minimize this gap, we design a novel structure that combines biological computation and DNNs. The V1-MT architecture we propose not only balances neurophysiological plausibility but also establishes a comprehensive model capable of replicating human multi-order motion perception in both experimental and complex natural settings. Coupled with our unsupervised learning method, the model is capable of learning first-order motion estimation with a tendency similar to supervised learning. Using this model as a biological agent, we also gain a scientific understanding of the functionality and meaning of higher-order motion perception observed in multiple species.

Despite these advancements, the current model has several limitations that open new avenues for future research.

7.2.1 Limits of Current Unsupervised Learning

Engineering-wise

Chapter 3 proposed several approaches for temporal modeling in unsupervised learning. However, these methods have limitations and areas for future improvement.

Firstly, the dynamic training enhancer introduces online scene transitions during training, increasing data processing time. Each training iteration requires five inferences, making the training phase approximately five times longer than the baseline. Secondly, the proposed temporal smoothness regularization (TSR) relies on short-time-window motion invariance, which

may be less effective in low-sampling-rate videos or fast-motion scenes. Using high-frame-rate cameras could reduce the video sampling window, potentially enhancing TSR’s effectiveness. This is a promising avenue given the accessibility of high-speed cameras compared to optical flow ground truths in natural settings. A new unsupervised learning protocol could use the high frame rate, allowing the model to learn motion rules in a slower world and gradually adapt to faster movements in a self-supervised manner. Thirdly, while the continuous movement of artificial occluders is simulated using the Markov property, superpixel algorithm-shaped image fragments do not fully capture natural objects. Future work could use instance segmentation networks to extract entire objects from scenes, simulating their motion more naturally. Additionally, the current method only simulates the 2D affine transformation of the artificial occluder, which does not capture the full complexity of the 3D world. A more realistic simulation could further improve performance.

Currently, the unsupervised learning protocol is mainly based on image-warping and reconstruction loss. However, these methods have inherent drawbacks. The reconstruction loss is not robust enough to cope with content variation and occlusion, especially in complex optical environments. Although several robust loss functions, such as SSIM and Census, have been developed to avoid brightness variation, they do not fully solve this problem [159]. Therefore, more robust methods are needed, such as constraints from multi-frame extension [53, 71, 168], epipolar constraints [198], 3D geometrical constraints [191, 200], and stereo depth [86, 181].

Scientific-wise

Unsupervised learning conceptually provides a promising approach to simulating animal behavior by allowing motion learning in a natural environment. However, current methods still do not capture the full spectrum of animal behavior in learning. Unsupervised learning, using the image-warp and reconstruction method, infers optical flow to match pixel correspondence based on brightness. However, in nature, organisms learn motion perception through interactions with their environment. This process involves integrating multiple sensory inputs and the brain’s complex processing mechanisms. Initially, organisms rely on various senses, such as vision, hearing, and touch, to perceive motion. For example, the visual system captures changes in light through photoreceptors in the retina, which then transmit neural signals to the brain for processing. Specific areas in the brain, such as the primary visual cortex and motor cortex, are responsible for processing and integrating sensory inputs. These areas learn and adapt to motion perception through neuron interactions and synaptic plasticity.

Through continuous experience and interaction with their environment, organisms can improve their motion perception abilities. For instance, a baby learns motion perception and coordination through activities like crawling and walking. Organisms use feedback mechanisms to self-correct and improve. For example, when a person reaches out to grab an object, visual and tactile feedback help adjust the hand’s position and grip strength to successfully grasp the

target.

Considering these issues, integrating unsupervised learning with Reinforcement Learning (RL) might better model the learning processes of organisms. RL emphasizes gaining experience through interactions with the environment and optimizing behavior strategies through reward and punishment mechanisms, similar to how organisms learn and adapt to their environment through trial and error and feedback mechanisms. In RL, an agent influences the environment by taking actions and adjusting its strategies based on environmental feedback (rewards or punishments). This process resembles how organisms optimize their motion perception and behavior strategies through sensory feedback and behavioral adjustments. Therefore, applying RL to simulate the learning process of organisms might more accurately reflect how organisms adapt and optimize their behavior compared to unsupervised learning. Future research can further explore integrating multisensory inputs and complex neural network structures to more accurately simulate biological learning and adaptation processes. This not only helps understand biological learning mechanisms but also potentially enhances the adaptability and intelligence of AI in dynamic environments.

7.2.2 Unclear Neural Implementation

While we show that the attention-based recurrent network is a promising computational model of human visual motion grouping and segmentation, how its complex architecture (where all responses can be influenced by all other responses) is actually implemented in the human brain remains an open question. Especially for self-attention mechanisms based on motion integration, although it successfully captured the function of the MT area in motion integration, the dynamic graph construction assumes all responses can be influenced by all other responses, depending on the response similarity. That is, dynamically, any response integrator can be connected to any other, no matter how far apart they are spatially. This is what allows the network to integrate local motion into global motion, but how this could be implemented neurally is very unclear. Additionally, we do not take into account the benefits of biological processing, such as energy efficiency.

In Chapter 6, we proposed that the dual-channel structure is also a conceptually simplified version of the real neural implementation. In particular, we employ a simplistic, linear fusion approach in our model to merge first-order and higher-order channels, while the interaction between first- and second-order signals remains controversial in the current study and may be much more complicated for the real neural system [100]. A further reasonable explanation of the neural implementation or an alternative, more neurally plausible implementation needs to be investigated in the future.

7.2.3 Blacks Box Problem Still Exists

In Chapter 5, while we attempted to incorporate computational constraints such as motion energy computation to enhance the model’s transparency and explainability, it is inevitable that the black box problem remains in the current model. For example, in Stage II, we leverage the attention mechanism and build a recurrent process to integrate and segregate the motion field. However, it is still unclear how the neural networks achieve this adaptive integration ability.

More specifically, we utilized visualization methods to show how local regions are interconnected during the integration process, but it remains unclear how the initial motion energy features are merged in the convolution layers to generate these connections, and how they identify objects and occlusions in complex scenarios. To increase explainability, we used the inverse max activation method [192] to demonstrate the preferred stimuli of each unit from Stage I and the deep layers of Stage II. This allowed us to understand the general mechanism carried out by the motion model: the units in Stage I prefer motion with simple frequency components like drifting gratings, while the second-stage unclassified units respond to more complex motion patterns consisting of both central and background motions, as shown at the bottom of Fig. 5.5 (B). This suggests that the classical classification of motion neurons into component and pattern cells might be insufficient to characterize the motion integration properties of these neurons. Such neural patterns are difficult to explain using current methodologies.

Similarly, in Chapter 6, we integrated multi-layer 3D CNNs to serve the higher-order motion channel. Although the 3D CNN with ReLU nonlinearity is powerful for extracting second-order motion features, it inevitably compromises the model’s transparency and explainability compared to the motion energy model. It is challenging to interpret which high-order features are being extracted and utilized to infer optical flow, such as long-term temporal statistics, depth cues, occlusions, structural constraints, or layer estimation. In fact, these features surpass mere motion perception and touch upon the integration of the broader functions of the human vision system.

It is promising to develop more advanced technologies and analytical methods to understand the inner mechanisms of DNNs. Future research could explore the use of techniques like explainable AI (XAI) [40], which focuses on making the operations of DNNs more interpretable to humans. Methods such as layer-wise relevance propagation (LRP) [7], integrated gradients, and attention visualization can provide insights into how specific features are processed and combined to produce the network’s output. Additionally, hybrid models that combine interpretable physical models with DNNs could offer a balance between performance and explainability. Continued advancements in these areas will be crucial for demystifying the black-box nature of neural networks and enhancing their applicability in complex real-world scenarios.

7.2.4 Vision Functionality is Interconnected

Despite our efforts to model and analyze human motion perception as an independent system, the reality is that the motion perception system in the human brain is not a standalone area. It primarily involves regions traversed by the dorsal stream and is closely connected with other brain regions. At the perceptual level, motion is intricately linked with other high-level visual functions such as object segmentation, self-motion, and depth perception, with motion perception influenced by these higher-order cues and information. Furthermore, human motion perception is not limited to the 2D image plane; it extends to 3D space, integrating depth and stereoscopic cues to infer the true motion of objects. Broadly speaking, these cues go beyond the scope of classical Fourier motion research and belong to a class of higher-order motion, similar to second-order motion. In Chapter 6, we also employed 3D-CNNs as feature extractors for this type of motion. However, it is challenging to interpret which high-order features are being extracted and utilized to infer optical flow, such as long-term temporal statistics, depth cues, occlusions, structural constraints, or layer estimation. These features surpass mere motion perception and touch upon the integration of the broader functions of the human vision system [123].

Overall, the results of this thesis, especially those in Chapter 6, underscore that human motion perception is not solely dependent on "motion" per se. It involves intrinsic interactions with higher-level vision functions beyond simple brightness shifts. The human vision system has evolved specialized channels and mechanisms to handle this complex information, incorporating separate first-order and higher-order motion processing areas [27, 83], along with the ventral and dorsal streams for distinct functionalities. Insights from such biological concepts have influenced numerous computer vision works, such as the Two-Stream Net [153]. Similarly, our dual-channel strategy demonstrates promising results in extending basic first-order motion perception to include higher-order motion perception, which is closely related to other vision functions like depth estimation, layering, and segmentation. It is anticipated that a foundational vision model that links multiple vision functions will continue to be a progressive direction for better understanding and simulating the human vision system.

In conclusion, this dissertation lays the groundwork for future research by highlighting the necessity of integrating multiple vision functions to achieve a comprehensive understanding of motion perception. Future studies should aim to develop more advanced models that incorporate high-order features and leverage the strengths of both supervised and unsupervised learning techniques. Additionally, exploring the application of reinforcement learning could provide deeper insights into how organisms adapt and optimize their behavior through environmental interactions, potentially leading to more robust and interpretable AI systems. Expanding on the dual-channel approach and incorporating multisensory inputs will be crucial for advancing the field and creating models that more accurately reflect the complexities of the human visual system.

Bibliography

- [1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012.
- [2] E. H. Adelson and J. R. Bergen. Spatiotemporal energy models for the perception of motion. *Josa a*, 2(2):284–299, 1985.
- [3] T. D. Albright. Form-cue invariant motion processing in primate visual cortex. *Science*, 255(5048):1141–1143, 1992.
- [4] K. Amano, M. Edwards, D. R. Badcock, and S. Nishida. Adaptive pooling of visual motion signals by the human visual system revealed with a novel multi-element stimulus. *Journal of vision*, 9(3):4–4, 2009.
- [5] S. M. Anstis and B. J. Rogers. Illusory reversal of visual depth and movement during changes of contrast. *Vision Research*, 15(8):957–IN6, 1975.
- [6] S. M. Anstis and B. J. Rogers. Illusory continuous motion from oscillating positive-negative patterns: Implications for motion perception. *Perception*, 15(5):627–640, 1986.
- [7] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):e0130140, 2015.
- [8] D. R. Badcock and A. M. Derrington. Detecting the displacement of periodic patterns. *Vision Research*, 25(9):1253–1258, 1985.
- [9] C. L. Baker and I. Mareschal. Processing of second-order stimuli in the visual cortex. *Progress in brain research*, 134:171–191, 2001.
- [10] C. L. Baker Jr. Central neural mechanisms for detecting second-order motion. *Current opinion in neurobiology*, 9(4):461–466, 1999.
- [11] H. Barlow, R. Hill, and W. Levick. Retinal ganglion cells responding selectively to direction and speed of image motion in the rabbit. *The Journal of physiology*, 173(3):377, 1964.

- [12] H. Barlow and W. R. Levick. The mechanism of directionally selective units in rabbit's retina. *The Journal of physiology*, 178(3):477, 1965.
- [13] J. L. Barron, D. J. Fleet, and S. S. Beauchemin. Performance of optical flow techniques. *International journal of computer vision*, 12:43–77, 1994.
- [14] Blender Online Community. Blender - a 3d modelling and rendering package, 2 2021.
- [15] J.-Y. Bouguet et al. Pyramidal implementation of the affine lucas kanade feature tracker description of the algorithm. *Intel corporation*, 5(1-10):4, 2001.
- [16] R. N. Bracewell and R. N. Bracewell. *The Fourier transform and its applications*, volume 31999. McGraw-Hill New York, 1986.
- [17] R. O. Brown and S. He. Visual motion of missing-fundamental patterns: motion energy versus feature correspondence. *Vision Research*, 40(16):2135–2147, 2000.
- [18] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, Oct. 2012.
- [19] M. Carandini and D. J. Heeger. Summation and division by neurons in primate visual cortex. *Science*, 264(5163):1333–1336, 1994.
- [20] E. Castet and J. Zanker. Long-range interactions in the spatial integration of motion signals. *Spatial Vision*, 12(3):287–307, 1999.
- [21] P. Cavanagh and G. Mather. Motion: the long and short of it. *Spatial vision*, 4(2-3):103–129, 1989.
- [22] S. Celebrini and W. T. Newsome. Microstimulation of extrastriate area mst influences performance on a direction discrimination task. *Journal of Neurophysiology*, 73(2):437–448, 1995.
- [23] K. Cho, B. van Merriënboer, Ç. Gülçehre, F. Bougares, H. Schwenk, and Y. Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *CoRR*, abs/1406.1078, 2014.
- [24] C. Chubb, J. McGowan, G. Sperling, and P. Werkhoven. Non-fourier motion analysis. In *Ciba Foundation Symposium 184-Higher-Order Processing in the Visual System: Higher-Order Processing in the Visual System: Ciba Foundation Symposium 184*, pages 193–210. Wiley Online Library, 2007.
- [25] C. Chubb and G. Sperling. Drift-balanced random stimuli: a general basis for studying non-fourier motion perception. *JOSA A*, 5(11):1986–2007, 1988.

- [26] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.
- [27] C. W. Clifford, J. N. Freedman, and L. M. Vaina. First-and second-order motion perception in gabor micropattern stimuli: psychophysics and computational modelling. *Cognitive brain research*, 6(4):263–271, 1998.
- [28] C. W. Clifford, J. N. Freedman, and L. M. Vaina. First-and second-order motion perception in gabor micropattern stimuli: psychophysics and computational modelling. *Cognitive brain research*, 6(4):263–271, 1998.
- [29] C. W. Clifford and L. M. Vaina. A computational model of selective deficits in first and second-order motion processing. *Vision Research*, 39(1):113–130, 1999.
- [30] E. Coumans and Y. Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning, 4 2016.
- [31] J. G. Daugman and C. J. Downing. Demodulation, predictive coding, and spatial vision. *JOSA A*, 12(4):641–660, 1995.
- [32] D. B. de Jong, F. Paredes-Vallés, and G. C. de Croon. How do neural networks estimate optical flow? a neuropsychology-inspired study. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8290–8305, 2021.
- [33] G. C. DeAngelis, I. Ohzawa, and R. Freeman. Spatiotemporal organization of simple-cell receptive fields in the cat’s striate cortex. i. general characteristics and postnatal development. *Journal of neurophysiology*, 69(4):1091–1117, 1993.
- [34] G. C. DeAngelis, I. Ohzawa, and R. D. Freeman. Receptive-field dynamics in the central visual pathways. *Trends in neurosciences*, 18(10):451–458, 1995.
- [35] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- [36] A. M. Derrington and D. R. Badcock. Separate detectors for simple and complex grating patterns? *Vision research*, 25(12):1869–1878, 1985.
- [37] 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.
- [38] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. Van Der Smagt, D. Cremers, and T. Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015.

- [39] A. Dosovitskiy, P. Fischer, E. Ilg, P. Häusser, C. Hazırbaş, V. Golkov, P. v.d. Smagt, D. Cremers, and T. Brox. FlowNet: Learning optical flow with convolutional networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.
- [40] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, et al. Explainable ai (xai): Core ideas, techniques, and solutions. *ACM Computing Surveys*, 55(9):1–33, 2023.
- [41] S. Exner. Experimentelle untersuchung der einfachsten psychischen processe. *Archiv für die gesamte Physiologie des Menschen und der Tiere*, 11(1):403–432, 1875.
- [42] G. Farnebäck. Two-frame motion estimation based on polynomial expansion. In *Image Analysis: 13th Scandinavian Conference, SCIA 2003 Halmstad, Sweden, June 29–July 2, 2003 Proceedings 13*, pages 363–370. Springer, 2003.
- [43] C. L. Fennema and W. B. Thompson. Velocity determination in scenes containing several moving objects. *Computer graphics and image processing*, 9(4):301–315, 1979.
- [44] C. L. Fennema and W. B. Thompson. Velocity determination in scenes containing several moving objects. *Computer graphics and image processing*, 9(4):301–315, 1979.
- [45] N. Fisher and J. M. Zanker. The directional tuning of the barber-pole illusion. *Perception*, 30(11):1321–1336, 2001.
- [46] O. Freifeld, S. Hauberg, K. Batmanghelich, and J. W. Fisher. Transformations based on continuous piecewise-affine velocity fields. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2496–2509, 2017.
- [47] A. Geiger, P. Lenz, and R. Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.
- [48] M. A. Georgeson and T. M. Shackleton. Monocular motion sensing, binocular motion perception. *Vision Research*, 29(11):1511–1523, 1989.
- [49] J. J. Gibson. The perception of the visual world. 1950.
- [50] S. Gilaie-Dotan. Visual motion serves but is not under the purview of the dorsal pathway. *Neuropsychologia*, 89:378–392, 2016.
- [51] P. Godet, A. Boulch, A. Plyer, and G. Le Besnerais. Starflow: A spatiotemporal recurrent cell for lightweight multi-frame optical flow estimation. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 2462–2469. IEEE, 2021.

- [52] K. Greff, F. Belletti, L. Beyer, C. Doersch, Y. Du, D. Duckworth, D. J. Fleet, D. Gnanaprasam, F. Golemo, C. Herrmann, T. Kipf, A. Kundu, D. Lagun, I. Laradji, H.-T. D. Liu, H. Meyer, Y. Miao, D. Nowrouzezahrai, C. Oztireli, E. Pot, N. Radwan, D. Rebain, S. Sabour, M. S. M. Sajjadi, M. Sela, V. Sitzmann, A. Stone, D. Sun, S. Vora, Z. Wang, T. Wu, K. M. Yi, F. Zhong, and A. Tagliasacchi. Kubric: a scalable dataset generator. 2022.
- [53] S. Guan, H. Li, and W.-S. Zheng. Unsupervised learning for optical flow estimation using pyramid convolution lstm. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pages 181–186. IEEE, 2019.
- [54] V. Guizilini, K.-H. Lee, R. Ambruş, and A. Gaidon. Learning optical flow, depth, and scene flow without real-world labels. *IEEE Robotics and Automation Letters*, 7(2):3491–3498, 2022.
- [55] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja. Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues. *Array*, 10:100057, 2021.
- [56] T. Handa and A. Mikami. Neuronal correlates of motion-defined shape perception in primate dorsal and ventral streams. *European journal of Neuroscience*, 48(10):3171–3185, 2018.
- [57] B. Hassenstein and W. Reichardt. Systemtheoretische analyse der zeit-, reihenfolgen- und vorzeichenauswertung bei der bewegungsperzeption des rüsselkäfers chlorophanus. *Zeitschrift für Naturforschung B*, 11(9-10):513–524, 1956.
- [58] D. J. Heeger. Modeling simple-cell direction selectivity with normalized, half-squared, linear operators. *Journal of neurophysiology*, 70(5):1885–1898, 1993.
- [59] G. B. Henning, B. G. Hertz, and D. Broadbent. Some experiments bearing on the hypothesis that the visual system analyses spatial patterns in independent bands of spatial frequency. *Vision research*, 15(8-9):887–897, 1975.
- [60] B. K. Horn and B. G. Schunck. Determining optical flow. *Artificial intelligence*, 17(1-3):185–203, 1981.
- [61] A. Hosni, C. Rhemann, M. Bleyer, C. Rother, and M. Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE transactions on pattern analysis and machine intelligence*, 35(2):504–511, 2012.
- [62] Z. Huang, X. Shi, C. Zhang, Q. Wang, K. C. Cheung, H. Qin, J. Dai, and H. Li. Flow-former: A transformer architecture for optical flow. In *Computer Vision–ECCV 2022*:

- 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, pages 668–685. Springer, 2022.
- [63] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat’s striate cortex. *The Journal of physiology*, 148(3):574, 1959.
 - [64] D. H. Hubel and T. N. Wiesel. Receptive fields and functional architecture in two non-striate visual areas (18 and 19) of the cat. *Journal of neurophysiology*, 28(2):229–289, 1965.
 - [65] T.-W. Hui, X. Tang, and C. C. Loy. Liteflownet: A lightweight convolutional neural network for optical flow estimation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8981–8989, 2018.
 - [66] J. Hur and S. Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.
 - [67] J. Hur and S. Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5754–5763, 2019.
 - [68] 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 conference on computer vision and pattern recognition*, pages 2462–2470, 2017.
 - [69] W. Im, T.-K. Kim, and S.-E. Yoon. Unsupervised learning of optical flow with deep feature similarity. In *European Conference on Computer Vision*, pages 172–188. Springer, 2020.
 - [70] S. Ince and J. Konrad. Occlusion-aware optical flow estimation. *IEEE Transactions on Image Processing*, 17(8):1443–1451, 2008.
 - [71] J. Janai, F. Guey, A. Ranjan, M. Black, and A. Geiger. Unsupervised learning of multi-frame optical flow with occlusions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 690–706, 2018.
 - [72] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz. Super slo-mo: High quality estimation of multiple intermediate frames for video interpolation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 9000–9008, 2018.
 - [73] A. Johnston and C. Clifford. Perceived motion of contrast-modulated gratings: predictions of the multi-channel gradient model and the role of full-wave rectification. *Vision research*, 35(12):1771–1783, 1995.

- [74] A. Johnston, P. McOwan, and H. Buxton. A computational model of the analysis of some first-order and second-order motion patterns by simple and complex cells. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 250(1329):297–306, 1992.
- [75] J. P. Jones and L. A. Palmer. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *Journal of neurophysiology*, 58(6):1233–1258, 1987.
- [76] J. P. Jones, A. Stepnoski, and L. A. Palmer. The two-dimensional spectral structure of simple receptive fields in cat striate cortex. *Journal of Neurophysiology*, 58(6):1212–1232, 1987.
- [77] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova. What matters in unsupervised optical flow. In *European Conference on Computer Vision*, pages 557–572. Springer, 2020.
- [78] R. Jonschkowski, A. Stone, J. T. Barron, A. Gordon, K. Konolige, and A. Angelova. What matters in unsupervised optical flow. In *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part II 16*, pages 557–572. Springer, 2020.
- [79] A. Kendall, H. Martirosyan, S. Dasgupta, P. Henry, R. Kennedy, A. Bachrach, and A. Bry. End-to-end learning of geometry and context for deep stereo regression. In *Proceedings of the IEEE international conference on computer vision*, pages 66–75, 2017.
- [80] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [81] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [82] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.
- [83] T. Ledgeway and A. T. Smith. Evidence for separate motion-detecting mechanisms for first-and second-order motion in human vision. *Vision research*, 34(20):2727–2740, 1994.
- [84] T. Ledgeway and A. T. Smith. Evidence for separate motion-detecting mechanisms for first-and second-order motion in human vision. *Vision research*, 34(20):2727–2740, 1994.
- [85] E. LeDue, M. Zou, and N. Crowder. Spatiotemporal tuning in mouse primary visual cortex. *Neuroscience letters*, 528(2):165–169, 2012.
- [86] L. Liu, G. Zhai, W. Ye, and Y. Liu. Unsupervised learning of scene flow estimation fusing with local rigidity. In *IJCAI*, pages 876–882, 2019.

- [87] L. Liu, J. Zhang, R. He, Y. Liu, Y. Wang, Y. Tai, D. Luo, C. Wang, J. Li, and F. Huang. Learning by analogy: Reliable supervision from transformations for unsupervised optical flow estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6489–6498, 2020.
- [88] P. Liu, I. King, M. R. Lyu, and J. Xu. Ddflow: Learning optical flow with unlabeled data distillation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 8770–8777, 2019.
- [89] P. Liu, I. King, M. R. Lyu, and J. Xu. Ddflow: Learning optical flow with unlabeled data distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8770–8777, 2019.
- [90] P. Liu, M. Lyu, I. King, and J. Xu. Selfflow: Self-supervised learning of optical flow. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4571–4580, 2019.
- [91] S. Liu, K. Luo, N. Ye, C. Wang, J. Wang, and B. Zeng. Oiflow: occlusion-inpainting optical flow estimation by unsupervised learning. *IEEE Transactions on Image Processing*, 30:6420–6433, 2021.
- [92] 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*, pages 10012–10022, 2021.
- [93] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.
- [94] J. Lorenceau and M. Shiffrar. The influence of terminators on motion integration across space. *Vision research*, 32(2):263–273, 1992.
- [95] W. Lotter, G. Kreiman, and D. Cox. Deep predictive coding networks for video prediction and unsupervised learning. *arXiv preprint arXiv:1605.08104*, 2016.
- [96] Z.-L. Lu and G. Sperling. Contrast gain control in first-and second-order motion perception. *JOSA A*, 13(12):2305–2318, 1996.
- [97] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *IJCAI’81: 7th international joint conference on Artificial intelligence*, volume 2, pages 674–679, 1981.
- [98] A. Luo, F. Yang, K. Luo, X. Li, H. Fan, and S. Liu. Learning optical flow with adaptive graph reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 1890–1898, 2022.

- [99] K. Luo, C. Wang, S. Liu, H. Fan, J. Wang, and J. Sun. Upflow: Upsampling pyramid for unsupervised optical flow learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1045–1054, 2021.
- [100] K. Maruya and S. Nishida. Spatial pooling of one-dimensional second-order motion signals. *Journal of Vision*, 10(13):24–24, 2010.
- [101] G. Mather. First-order and second-order visual processes in the perception of motion and tilt. *Vision research*, 1991.
- [102] G. Mather and L. Murdoch. Evidence for global motion interactions between first-order and second-order stimuli. *Perception*, 27(7):761–767, 1998.
- [103] G. Mather and S. West. Evidence for second-order motion detectors. *Vision research*, 1993.
- [104] N. Mayer, E. Ilg, P. Fischer, C. Hazirbas, D. Cremers, A. Dosovitskiy, and T. Brox. What makes good synthetic training data for learning disparity and optical flow estimation? *International Journal of Computer Vision*, 126(9):942–960, 2018.
- [105] N. Mayer, E. Ilg, P. Häusser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. arXiv:1512.02134.
- [106] M. Mazurek, M. Kager, and S. D. Van Hooser. Robust quantification of orientation selectivity and direction selectivity. *Frontiers in neural circuits*, 8:92, 2014.
- [107] C. McCool and K. Britten. 2.10 - cortical processing of visual motion. In *The Senses: A Comprehensive Reference*, pages 157–187. Academic Press, New York, 2008.
- [108] J. McDermott, Y. Weiss, and E. H. Adelson. Beyond junctions: nonlocal form constraints on motion interpretation. *Perception*, 30(8):905–923, 2001.
- [109] P. Mehrani and J. K. Tsotsos. Self-attention in vision transformers performs perceptual grouping, not attention. *arXiv preprint arXiv:2303.01542*, 2023.
- [110] S. Meister, J. Hur, and S. Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.
- [111] S. Meister, J. Hur, and S. Roth. Unflow: Unsupervised learning of optical flow with a bidirectional census loss. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

- [112] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3061–3070, 2015.
- [113] M. Menze and A. Geiger. Object scene flow for autonomous vehicles. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.
- [114] M. Menze, C. Heipke, and A. Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015.
- [115] P. Mineault, S. Bakhtiari, B. Richards, and C. Pack. Your head is there to move you around: Goal-driven models of the primate dorsal pathway. *Advances in Neural Information Processing Systems*, 34:28757–28771, 2021.
- [116] J. Movshon, E. Adelson, M. Gizzi, and W. Newsome. The analysis of moving visual patterns 1986. in pattern recognition mechanisms, edited by c. chagas, r. gattass, and c. gross. new york: Springer-verlag. *Frontiers in Cognitive Neuroscience*, page 148, 1995.
- [117] J. Movshon, E. Adelson, M. Gizzi, and W. T. Newsome. The analysis of moving visual patterns. In *Frontiers in cognitive neuroscience*. MIT Press, 1992.
- [118] D. Nakamura and H. Gomi. Decoding self-motion from visual image sequence predicts distinctive features of reflexive motor responses to visual motion. *Neural Networks*, 162:516–530, 2023.
- [119] M. Neoral, J. Šochman, and J. Matas. Continual occlusion and optical flow estimation. In *Asian Conference on Computer Vision*, pages 159–174. Springer, 2018.
- [120] S. Nishida. Spatiotemporal properties of motion perception for random-check contrast modulations. *Vision Research*, 33(5-6):633–645, 1993.
- [121] S. Nishida. Advancement of motion psychophysics: Review 2001–2010. *Journal of vision*, 11(5):11–11, 2011.
- [122] S. Nishida, H. Ashida, and T. Sato. Complete interocular transfer of motion aftereffect with flickering test. *Vision research*, 34(20):2707–2716, 1994.
- [123] S. Nishida, T. Kawabe, M. Sawayama, and T. Fukiage. Motion perception: From detection to interpretation. *Annual review of vision science*, 4:501–523, 2018.
- [124] S. Nishida and T. Sato. Positive motion after-effect induced by bandpass-filtered random-dot kinematograms. *Vision research*, 32(9):1635–1646, 1992.
- [125] S. Nishida and T. Sato. Motion aftereffect with flickering test patterns reveals higher stages of motion processing. *Vision research*, 35(4):477–490, 1995.

- [126] A. Noest and A. Van Den Berg. The role of early mechanisms in motion transparency and coherence. *Spatial Vision*, 7(2):125–147, 1993.
- [127] U. S. N. B. of Standards and F. E. Nicodemus. *Geometrical considerations and nomenclature for reflectance*, volume 160. US Department of Commerce, National Bureau of Standards Washington, DC, USA, 1977.
- [128] L. P. O’KEEFE and J. A. Movshon. Processing of first-and second-order motion signals by neurons in area mt of the macaque monkey. *Visual neuroscience*, 15(2):305–317, 1998.
- [129] C. Pack and R. Born. Cortical mechanisms for the integration of visual motion. *The senses: A comprehensive reference*, pages 189–218, 2008.
- [130] C. C. Pack and R. T. Born. Temporal dynamics of a neural solution to the aperture problem in visual area mt of macaque brain. *Nature*, 409(6823):1040–1042, 2001.
- [131] V. C. Paulun, T. Kawabe, S. Nishida, and R. W. Fleming. Seeing liquids from static snapshots. *Vision research*, 115:163–174, 2015.
- [132] J. Peirce, J. Gray, S. Simpson, M. MacAskill, R. Höchenberger, H. Sogo, E. Kastman, and J. Lindeløv. Psychopy2: Experiments in behavior made easy. *Behavior Research Methods*, 2019.
- [133] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Computer Vision and Pattern Recognition*, 2016.
- [134] J. A. Perrone. A visual motion sensor based on the properties of v1 and mt neurons. *Vision research*, 44(15):1733–1755, 2004.
- [135] J. A. Perrone and A. Thiele. Speed skills: measuring the visual speed analyzing properties of primate mt neurons. *Nature neuroscience*, 4(5):526–532, 2001.
- [136] N. J. Priebe, C. R. Cassanello, and S. G. Lisberger. The neural representation of speed in macaque area mt/v5. *Journal of Neuroscience*, 23(13):5650–5661, 2003.
- [137] N. J. Priebe, C. R. Cassanello, and S. G. Lisberger. The neural representation of speed in macaque area mt/v5. *Journal of Neuroscience*, 23(13):5650–5661, 2003.
- [138] N. J. Priebe, S. G. Lisberger, and J. A. Movshon. Tuning for spatiotemporal frequency and speed in directionally selective neurons of macaque striate cortex. *Journal of Neuroscience*, 26(11):2941–2950, 2006.
- [139] A. Ranjan and M. J. Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017.

- [140] A. Ranjan and M. J. Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4161–4170, 2017.
- [141] A. Ranjan, J. Janai, A. Geiger, and M. J. Black. Attacking optical flow. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2404–2413, 2019.
- [142] R. P. Rao and D. H. Ballard. Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature neuroscience*, 2(1):79–87, 1999.
- [143] R. Rideaux and A. E. Welchman. But still it moves: static image statistics underlie how we see motion. *Journal of Neuroscience*, 40(12):2538–2552, 2020.
- [144] R. Rideaux and A. E. Welchman. Exploring and explaining properties of motion processing in biological brains using a neural network. *Journal of Vision*, 21(2):11–11, 2021.
- [145] D. L. Ringach. Spatial structure and symmetry of simple-cell receptive fields in macaque primary visual cortex. *Journal of neurophysiology*, 88(1):455–463, 2002.
- [146] C. D. Salzman, C. M. Murasugi, K. H. Britten, and W. T. Newsome. Microstimulation in visual area mt: effects on direction discrimination performance. *Journal of Neuroscience*, 12(6):2331–2355, 1992.
- [147] J. Schmalfuss, L. Mehl, and A. Bruhn. Distracting downpour: Adversarial weather attacks for motion estimation. In *Proc. International Conference on Computer Vision (ICCV)*. IEEE/CVF, 2023.
- [148] A. J. Schofield. What does second-order vision see in an image? *Perception*, 29(9):1071–1086, 2000.
- [149] T. Serre. Deep learning: the good, the bad, and the ugly. *Annual review of vision science*, 5:399–426, 2019.
- [150] C. E. Shannon. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423, 1948.
- [151] X. Shi, Z. Huang, W. Bian, D. Li, M. Zhang, K. C. Cheung, S. See, H. Qin, J. Dai, and H. Li. Videoflow: Exploiting temporal cues for multi-frame optical flow estimation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12469–12480, 2023.
- [152] E. P. Simoncelli and D. J. Heeger. A model of neuronal responses in visual area mt. *Vision research*, 38(5):743–761, 1998.

- [153] K. Simonyan and A. Zisserman. Two-stream convolutional networks for action recognition in videos. *Advances in neural information processing systems*, 27, 2014.
- [154] A. T. Smith. Correspondence-based and energy-based detection of second-order motion in human vision. *JOSA A*, 11(7):1940–1948, 1994.
- [155] F. Solari, M. Chessa, N. K. Medathati, and P. Kornprobst. What can we expect from a v1-mt feedforward architecture for optical flow estimation? *Signal Processing: Image Communication*, 39:342–354, 2015.
- [156] J. A. Solomon and G. Sperling. Full-wave and half-wave rectification in second-order motion perception. *Vision research*, 34(17):2239–2257, 1994.
- [157] G. Sperling. Movement perception in computer-driven visual displays. *Behavior Research Methods & Instrumentation*, 8:144–151, 1976.
- [158] G. Sperling. Three stages and two systems of visual processing. *Spatial vision*, 4(2-3):183–207, 1989.
- [159] A. Stone, D. Maurer, A. Ayvaci, A. Angelova, and R. Jonschkowski. Smurf: Self-teaching multi-frame unsupervised raft with full-image warping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3887–3896, 2021.
- [160] L. Stone, A. Watson, and J. Mulligan. Effect of contrast on the perceived direction of a moving plaid. *Vision research*, 30(7):1049–1067, 1990.
- [161] K. Storrs, O. Kampman, R. Rideaux, G. Maiello, and R. Fleming. Properties of v1 and mt motion tuning emerge from unsupervised predictive learning. *Journal of Vision*, 22(14):4415–4415, 2022.
- [162] D. Stutz, A. Hermans, and B. Leibe. Superpixels: An evaluation of the state-of-the-art. *CoRR*, abs/1612.01601, 2016.
- [163] D. Sun, S. Roth, and M. J. Black. Secrets of optical flow estimation and their principles. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 2432–2439. IEEE, 2010.
- [164] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.
- [165] D. Sun, X. Yang, M.-Y. Liu, and J. Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.

- [166] P. Sun, C. Chubb, and G. Sperling. Two mechanisms that determine the barber-pole illusion. *Vision research*, 111:43–54, 2015.
- [167] Z. Sun, Y.-J. Chen, Y.-H. Yang, and S. Nishida. Comparative analysis of visual motion perception: Computer vision models versus human vision. In *Conference on Cognitive Computational Neuroscience*, Oxford, UK, August 24-27 2023.
- [168] Z. Sun, Z. Luo, and S. Nishida. Decoupled spatiotemporal adaptive fusion network for self-supervised motion estimation. *Neurocomputing*, 534:133–146, 2023.
- [169] Z. Sun, Z. Luo, and S. Nishida. Unsupervised learning of optical flow in a multi-frame dynamic environment using temporal dynamic modeling. *Complex & Intelligent Systems*, 10(2):2215–2231, 2024.
- [170] A. Sutter, G. Sperling, and C. Chubb. Measuring the spatial frequency selectivity of second-order texture mechanisms. *Vision research*, 35(7):915–924, 1995.
- [171] Z. Teed and J. Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European conference on computer vision*, pages 402–419. Springer, 2020.
- [172] J. C. Theobald, B. J. Duistermars, D. L. Ringach, and M. A. Frye. Flies see second-order motion. *Current Biology*, 18(11):R464–R465, 2008.
- [173] L. Tian, Z. Tu, D. Zhang, J. Liu, B. Li, and J. Yuan. Unsupervised learning of optical flow with cnn-based non-local filtering. *IEEE Transactions on Image Processing*, 29:8429–8442, 2020.
- [174] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Sixth international conference on computer vision (IEEE Cat. No. 98CH36271)*, pages 839–846. IEEE, 1998.
- [175] J. K. Tsotsos, Y. Liu, J. C. Martinez-Trujillo, M. Pomplun, E. Simine, and K. Zhou. Attending to visual motion. *Computer Vision and Image Understanding*, 100(1-2):3–40, 2005.
- [176] L. M. Vaina, S. A. Beardsley, and S. K. Rushton. *Optic flow and beyond*, volume 324. Springer Science & Business Media, 2004.
- [177] L. M. Vaina, A. Cowey, and D. Kennedy. Perception of first-and second-order motion: Separable neurological mechanisms? *Human brain mapping*, 7(1):67–77, 1999.
- [178] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

- [179] L. Wang, Y. Guo, L. Liu, Z. Lin, X. Deng, and W. An. Deep video super-resolution using hr optical flow estimation. *IEEE Transactions on Image Processing*, 29:4323–4336, 2020.
- [180] X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.
- [181] Y. Wang, P. Wang, Z. Yang, C. Luo, Y. Yang, and W. Xu. Unos: Unified unsupervised optical-flow and stereo-depth estimation by watching videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8071–8081, 2019.
- [182] Y. Wang, Y. Yang, Z. Yang, L. Zhao, P. Wang, and W. Xu. Occlusion aware unsupervised learning of optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4884–4893, 2018.
- [183] Y. Wei, H. Xiao, H. Shi, Z. Jie, J. Feng, and T. S. Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7268–7277, 2018.
- [184] Y. Weiss, E. P. Simoncelli, and E. H. Adelson. Motion illusions as optimal percepts. *Nature neuroscience*, 5(6):598–604, 2002.
- [185] H. R. Wilson, V. P. Ferrera, and C. Yo. A psychophysically motivated model for two-dimensional motion perception. *Visual neuroscience*, 9(1):79–97, 1992.
- [186] H. Xu and J. Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1959–1968, 2020.
- [187] H. Xu, J. Zhang, J. Cai, H. Rezatofighi, and D. Tao. Gmflow: Learning optical flow via global matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8121–8130, 2022.
- [188] D. L. Yamins and J. J. DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature neuroscience*, 19(3):356–365, 2016.
- [189] Y.-H. Yang, T. Fukiage, Z. Sun, and S. Nishida. Psychophysical measurement of perceived motion flow of naturalistic scenes. *iScience in press*, 2023.
- [190] J. Yates. Motion illusions. <https://jake.vision/blog/motion-illusions>, Dec 2020.
- [191] Z. Yin and J. Shi. Geonet: Unsupervised learning of dense depth, optical flow and camera pose. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1983–1992, 2018.

- [192] J. Yosinski, J. Clune, A. Nguyen, T. Fuchs, and H. Lipson. Understanding neural networks through deep visualization. *arXiv preprint arXiv:1506.06579*, 2015.
- [193] J. J. Yu, A. W. Harley, and K. G. Derpanis. Back to basics: Unsupervised learning of optical flow via brightness constancy and motion smoothness. In *European Conference on Computer Vision*, pages 3–10. Springer, 2016.
- [194] T. Yu, S. Kumar, A. Gupta, S. Levine, K. Hausman, and C. Finn. Gradient surgery for multi-task learning. *Advances in Neural Information Processing Systems*, 33:5824–5836, 2020.
- [195] S. Yuan, X. Sun, H. Kim, S. Yu, and C. Tomasi. Optical flow training under limited label budget via active learning. In *European Conference on Computer Vision*, pages 410–427. Springer, 2022.
- [196] G. Zaai, R. Tuytel, R. Cilliers, J. R. Cock, A. Mischok, S. Majboroda, and J. Burger. Polyhaven: A curated public asset library for visual effects artists and game designers. 2021.
- [197] J. M. Zanker. Theta motion: a paradoxical stimulus to explore higher order motion extraction. *Vision research*, 33(4):553–569, 1993.
- [198] Y. Zhong, P. Ji, J. Wang, Y. Dai, and H. Li. Unsupervised deep epipolar flow for stationary or dynamic scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12095–12104, 2019.
- [199] Y.-X. Zhou and C. L. Baker Jr. A processing stream in mammalian visual cortex neurons for non-fourier responses. *Science*, 261(5117):98–101, 1993.
- [200] Y. Zou, Z. Luo, and J.-B. Huang. Df-net: Unsupervised joint learning of depth and flow using cross-task consistency. In *Proceedings of the European conference on computer vision (ECCV)*, pages 36–53, 2018.

List of Publications

Chapter Three

1. **Zitang Sun**, Zhengbo Luo, and Shin'ya Nishida. *Unsupervised learning of optical flow in a multi-frame dynamic environment using temporal dynamic modeling*. *Complex & Intelligent Systems*, 10(2):2215–2231, 2024.
 2. **Zitang Sun**, Zhengbo Luo, and Shin'ya Nishida. *Decoupled spatiotemporal adaptive fusion network for self-supervised motion estimation*. *Neurocomputing*, 534:133–146, 2023.
-

Chapter Four

1. **Zitang Sun**, Yen-Ju Chen, Yung-Hao Yang, and Shin'ya Nishida. *Comparative Analysis of Visual Motion Perception: Computer Vision Models versus Human Vision*. 2023 Conference on Cognitive Computational Neuroscience, 2023.
-

Chapter Five

1. **Zitang Sun**, Yen-Ju Chen, Yung-Hao Yang, and Shin'ya Nishida. *Modeling of Human Motion Perception Mechanism: A Simulation based on Deep Neural Network and Attention Transformer*. Poster Presentation, Annual Meeting of Vision Sciences Society (VSS), St. Pete Beach, U.S., May 2023.
 2. **Zitang Sun**, Yen-Ju Chen, Yung-Hao Yang, and Shin'ya Nishida. *Modeling Human Visual Motion Processing with Trainable Motion Energy Sensing and a Self-attention Network*. In *Advances in Neural Information Processing Systems*, volume 36, pages 24335–24348, 2023.
-

Chapter Six

1. **Zitang Sun**, Yen-Ju Chen, Yung-Hao Yang, and Shin'ya Nishida. *Acquisition of second-order motion perception by learning to recognize the motion of objects made by non-diffusive materials*. Talk Presentation, Annual Meeting of Vision Sciences Society (VSS), St. Pete Beach, U.S., May 2024.
-

Others

1. Yen-Ju Chen, **Zitang Sun**, and Shin'ya Nishida. *Feature-invariant processing of spatial segregation based on temporal asynchrony*. Journal of Vision, 24(5):15-15, 2024. DOI: [10.1167/jov.24.5.15](https://doi.org/10.1167/jov.24.5.15), URL: <https://doi.org/10.1167/jov.24.5.15>
2. Yen-Ju Chen, **Zitang Sun**, and Shin'ya Nishida. *Temporal Dynamics Gap between Position Tracking and Attribute Tracking*. Poster Presentation, Annual Meeting of Vision Sciences Society (VSS), St. Pete Beach, U.S., May 2024.
3. Yung-Hao Yang, Taiki Fukiage, **Zitang Sun**, and Shin'ya Nishida. *Psychophysical measurement of perceived motion flow of naturalistic scenes*. IScience, 26(12), 2023.
4. Yung-Hao Yang, Taiki Fukiage, **Zitang Sun**, and Shin'ya Nishida. *Temporal characteristics of perceived motion flow of naturalistic movies*. Poster Presentation, Annual Meeting of Vision Sciences Society (VSS), St. Pete Beach, U.S., May 2024.