High-Resolution Optical Flow from 1D Attention and Correlation

Haofei Xu[1*]   Jiaolong Yang[2]   Jianfei Cai[3]   Juyong Zhang[1]   Xin Tong[2]

[1]University of Science and Technology of China   [2]Microsoft Research Asia

[3]Department of Data Science and AI, Monash University

{xhf@mail., juyong@}ustc.edu.cn   {jiaoyan, xtong}@microsoft.com   jianfei.cai@monash.edu

Abstract

Optical flow is inherently a 2D search problem, and thus the computational complexity grows quadratically with respect to the search window, making large displacements matching infeasible for high-resolution images. In this paper, we take inspiration from Transformers and propose a new method for high-resolution optical flow estimation with significantly less computation. Specifically, a 1D attention operation is first applied in the vertical direction of the target image, and then a simple 1D correlation in the horizontal direction of the attended image is able to achieve 2D correspondence modeling effect. The directions of attention and correlation can also be exchanged, resulting in two 3D cost volumes that are concatenated for optical flow estimation. The novel 1D formulation empowers our method to scale to very high-resolution input images while maintaining competitive performance. Extensive experiments on Sintel, KITTI and real-world 4K (2160 × 3840) resolution images demonstrated the effectiveness and superiority of our proposed method. Code and models are available at https://github.com/haofeixu/flow1d.

1. Introduction

Optical flow estimation, a classic topic in computer vision, is a fundamental building block of various real-world applications such as 3D reconstruction [22], video processing [20] and action recognition [31]. The recent advancement of deep learning enables directly optical flow learning with a neural network [18]. By further improving the architectures and training strategies, deep learning based methods [18, 34, 35, 43, 37] have demonstrated stronger performance and faster inference speed compared with traditional optimization based approaches [13, 46, 33, 29, 44, 3].

An essential component in deep learning based optical flow frameworks is *cost volume* [14], which is usually computed by the dot product operation (also known as *correlation* [10]) between two feature vectors. It stores the match-

---
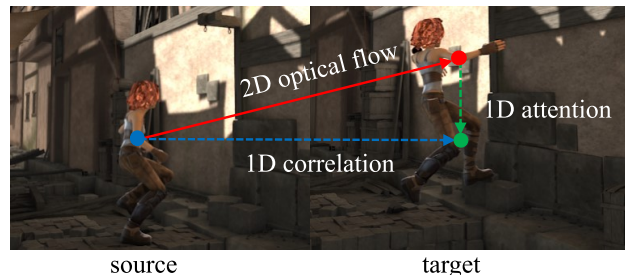*Work primarily done while interning at MSRA mentored by JY

Figure 1: Optical flow factorization. We factorize the 2D optical flow with 1D attention and correlation in orthogonal directions to achieve large displacements search on high-resolution images. Specifically, for the correspondence (red point) of the blue point, we first perform a 1D vertical attention to propagate the information of the red point to the green point, which lies on the same row of the blue point. Then a simple 1D correlation in the horizontal direction can be applied to build a horizontal cost volume. The vertical cost volume can be derived likewise with switched attention and correlation directions.

ing costs between each pixel in the source image and its potential correspondence candidates in the target image. By explicitly constructing a cost volume layer that encodes the search space, the network learns to better reason about the relative pixel displacements, as demonstrated by the superior performance of FlowNetC than FlowNetS that without such a layer [10, 18].

The original cost volume in FlowNetC [10] is constructed in a single scale and it has difficulty in modeling large displacements due to the quadratic complexities with respect to the search window. PWC-Net [34] migrates this problem by constructing multiple partial cost volumes in a coarse-to-fine warping framework. However, coarse-to-fine methods tend to miss small objects since they might not be visible in the highly-downsampled coarse scales, and thus have little chance to be correctly estimated [29, 32, 3]. Moreover, warping might introduce artifacts in occlusion regions [24], which may potentially hinder the network to learn correct correspondences.

Current state-of-the-art optical flow method, RAFT [37], maintains a single-resolution feature map and gradually estimates the flow updates in an iterative manner, eliminating several limitations of previous coarse-to-fine frameworks. One key component in RAFT is a 4D cost volume ($H \times W \times H \times W$) that is obtained by computing the correlations of all pairs. Thanks to such a large cost volume, RAFT achieves striking performance on established benchmarks. Nevertheless, the 4D cost volume requirement makes it difficult to scale to very high-resolution inputs due to the quadratic complexity with respect to the image resolution. Although one can partially alleviate this problem by processing on downsampled images, some fine-grained details, which might be critical for some scenarios (*e.g.*, ball sports and self-driving cars), will be inevitably lost in such a process. Furthermore, with the popularity of consumer-level high-definition cameras, it is much easier than before to get access to high-resolution videos, which accordingly raises the demand to be able to process such high-resolution videos with high efficiency.

To this end, we propose a new cost volume construction method for high-resolution optical flow estimation. Our key idea is to factorize the 2D search to two 1D substitutes in the vertical and horizontal direction, respectively, such that we can use 1D correlations to construct compact cost volumes.

Intuitively, such 1D correlations are not sufficient for optical flow estimation which is inherently a 2D search problem. However, as illustrated in Fig. 1, if we can propagate the information on the target image along the direction orthogonal to the correlation direction, the computed cost volume will contain meaningful correlation between the source pixel and its correspondence. This insight motivates us to design proper feature propagation and aggregation schemes for 1D correlation. Inspired by Transformers [38], we propose to learn such propagation with the attention mechanism, where we first apply 1D self attention on source feature (not shown in Fig. 1 for brevity), then 1D cross attention between the source and target features (see Fig. 3).

Our 1D formulation yields two 3D cost volumes of size ($H \times W \times W$) and ($H \times W \times H$), respectively, which are then concatenated for the subsequent optical flow regression. This way, we reduce the complexity of all-pair correlation [37] from $\mathcal{O}(H \times W \times H \times W)$ to $\mathcal{O}(H \times W \times (H+W))$, enabling our method to scale to very high-resolution inputs with significant less computation. For example, our method consumes $6\times$ less memory than RAFT on 1080p (1080×1920) videos. We also show flow results on real-world 4K (2160×3840) resolution images and our method can handle images more than 8K (4320×7680) resolution on a GPU with 32GB memory. Meanwhile, the evaluation on Sintel [4] and KITTI [26] shows that the accuracy of our method is only slightly worse than RAFT but outperforms other methods such as FlowNet2 [18] and PWC-Net [34].

Our contributions can be summarized as follows:

- We explore an innovative cost volume construction method which is fundamentally different from all existing methods.

- We show that cost volumes constructed using 1D correlations, despite somewhat counter-intuitive, can achieve striking flow estimation accuracy comparable to the state of the art.

- Our method is slightly inferior compared to RAFT in terms of accuracy but enjoys significantly less memory consumption, enabling us to process very high-resolution images (more than 8K resolution (4320×7680) in our experiment).

2. Related Work

Optical flow has traditionally been formulated as an optimization problem [13, 3]. A thorough comparison and evaluation of traditional methods can be found at [33]. In this section, we mainly review recent learning based methods from different perspectives.

Cost volume. The concept of cost volume dates back to stereo matching literature [14, 30]. A cost volume stores the matching costs for different pixel displacement candidates (*i.e.*, disparity in stereo) at each pixel coordinate. Thus it is a 3D tensor ($H \times W \times D$) in stereo matching, where D is the maximum disparity range. The cost volume serves as a discriminative representation of the search space and powerful optimization methods can be employed to filter outliers, and thus it usually leads to accurate results [12, 14].

The effectiveness of cost volume also benefits the optical flow community, either traditional [6, 42] or learning based methods [10, 34, 43, 24, 47, 37]. However, unlike 1D disparity in stereo, the search space in optical flow is 2D, thus resulting in a 4D cost volume ($H \times W \times (2R+1) \times (2R+1)$) for search radius R, which is computational expensive for large displacements. To alleviate this issue, a popular strategy is to use coarse-to-fine warping schemes [2, 34, 43, 47]. However, coarse-to-fine methods tend to miss fast-moving small objects [29, 24, 37]. Recently, RAFT [37] proposes to construct a 4D cost volume ($H \times W \times H \times W$) by computing all spatial correlations. Despite the state-of-the-art performance, RAFT is inherently constrained by the input resolution due to the quadratic complexity. In contrast, we propose to factorize the 2D optical flow with 1D attention and correlation, leading to significantly reduced complexities and allowing us to handle larger image resolutions.

Flow regression scheme. Top-performing learning based optical flow methods mostly share a similar structure: starting from a coarse flow estimate and then gradually refining the initial prediction [17]. They can be broadly classified into two categories: *non-iterative* and *iterative* meth-

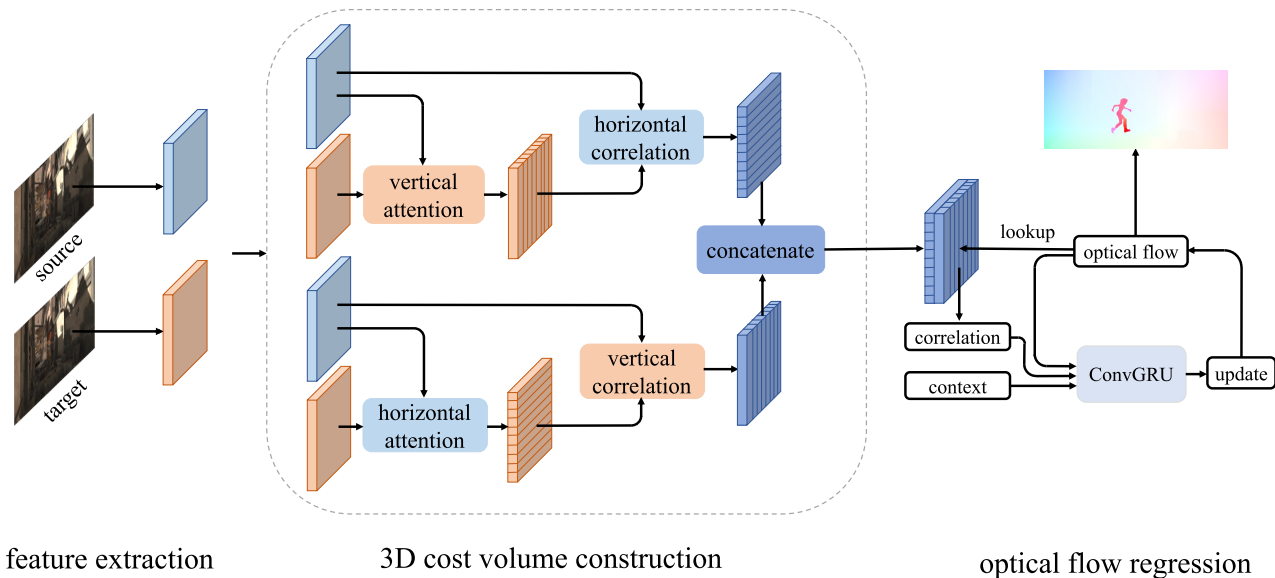| feature extraction | 3D cost volume construction | optical flow regression |

Figure 2: Overview of our framework. Given a pair of source and target images, we first extract $8\times$ downsampled features with a shared backbone network. The features are then used to construct two 3D cost volumes with *vertical attention, horizontal correlation* and *horizontal attention, vertical correlation*, respectively. The two cost volumes are then concatenated for optical flow regression, where we adopt RAFT's framework to estimate the optical flow in an iterative manner. After a number of iterations, the final optical flow prediction can be obtained. More details can be found in Sec. 4.

ods. Here, we use 'non-iterative' to denote that the refinement networks have their own trainable weights, while 'iterative' represents the network weights are shared at each refinement stage. Representative non-iterative frameworks include FlowNet2 [18], PWC-Net [34] and their variants [16, 45, 43, 47]. IRR-PWC [17] and RAFT [37] are two representative iterative methods. IRR-PWC shares a convolutional decoder to estimate flow in a coarse-to-fine framework with a limited number of iterations, while RAFT uses a ConvGRU [8] and applies this architecture for a large number of iterations (10+). In this paper, we validate the effectiveness of our proposed cost volume construction method with RAFT's framework due to its compactness and good performance. But in theory, our key idea is orthogonal to the flow regression scheme adopted.

Attention mechanism. Attention mechanism has achieved remarkable success in modeling long-range dependencies. However, it has one crucial limitation that the computational complexity grows quadratically with respect to the input size. This issue becomes severer when applying attention to vision tasks due to the large number of image pixels. Large volumes of work tries to reduce the complexity of attention by sparse connection patterns [7], low-rank approximations [40] or recurrent operations [9]. A thorough review of efficient attention mechanisms can be found at [36]. Among these methods, perhaps the most relevant to ours in vision are CCNet [15] and Axial-Deeplab [39], both of which use two 1D self attentions for global depen-

dency modeling. Different from the self attention in CC-Net and Axial-Deeplab, we use 1D cross attention and 1D correlation between a pair of source and target images to achieve large displacement correspondence search. Moreover, the output is a cost volume in our case to explicitly model matching costs while CCNet and Axial-Deeplab output a feature map.

3. Optical Flow Factorization

Optical flow is inherently a 2D search problem, but directly searching on the 2D image space is computationally intractable for very large displacements due to the quadratic complexity with respect to the search window. For example, the potential search space can be up to 10^4 pixels for search range $[-50, 50]$. This problem is yet more pronounced for high-resolution images.

To enable optical flow estimation on high-resolution images, we observe that the 2D search space can be approximated in two 1D directions. As illustrated in Fig. 1, to achieve 2D correspondence modeling effect between the blue and red points, we can first propagate the information of the red point to the green one, which lies on the same row of the blue point. Then a 1D search along the horizontal direction can capture the information of the red point.

In light of this, we propose to factorize the 2D optical flow with 1D attention [38] and 1D correlation [10] in orthogonal directions to achieve large displacements search on high-resolution images. Specifically, we first perform
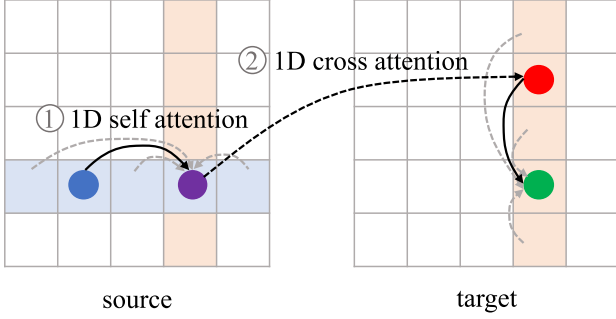
Figure 3: Illustration of self and cross attentions in the vertical aggregation process of target image feature. To properly propagate the information of red point (blue point's correspondence) onto the green point, we first perform a 1D horizontal self attention on the source image to make every point encode the information of its entire row (thus the purple point is aware of the blue point). Then a 1D vertical cross attention between the source and target is performed, where the green point receives the information of the red point *conditioned* on the purple point feature.

a 1D attention in the vertical direction, which computes a weighted combination of pixels that are in the same column. Then a simple 1D correlation in the horizontal direction can achieve 2D search effect. The resulting cost volume is 3D, similar to stereo methods [25, 41]. By exchanging the direction of attention and correlation, we can obtain another 3D cost volume. These two cost volumes are concatenated for optical flow regression (see Fig. 2).

Next, we present details of our *horizontal* cost volume construction with 1D vertical attention and 1D horizontal correlation. Its vertical counterpart can be similarly derived. For source and target images I_1 and I_2, a shared convolutional backbone is first used to extract features (see Fig. 2), we then construct the cost volume at the feature level.

3.1. 1D Attention

With source and target features $F_1, F_2 \in \mathbb{R}^{H \times W \times D}$, where H, W, D represents height, width and feature dimension, respectively, our goal is to generate a new feature $\hat{F}_2$ where every feature vector is aware of the features of points in F_2 lying in the same column. In this way, we can then perform a simple 1D search along the horizontal direction to achieve 2D correspondence modeling effect.

Concretely, we define $\hat{F}_2$ as a linear combination of column-wise features from F_2:

$$\hat{F}_2(h, w) = \sum_{i=0}^{H-1} f_{i,h,w} F_2(i, w), \qquad (1)$$

where $h = 0, 1, \cdots, H-1, w = 0, 1, \cdots, W-1, \hat{F}_2(h, w)$ is the output feature vector at position (h, w), $f_{i,h,w}$ is the combination weight. The key in our formulation lies in how
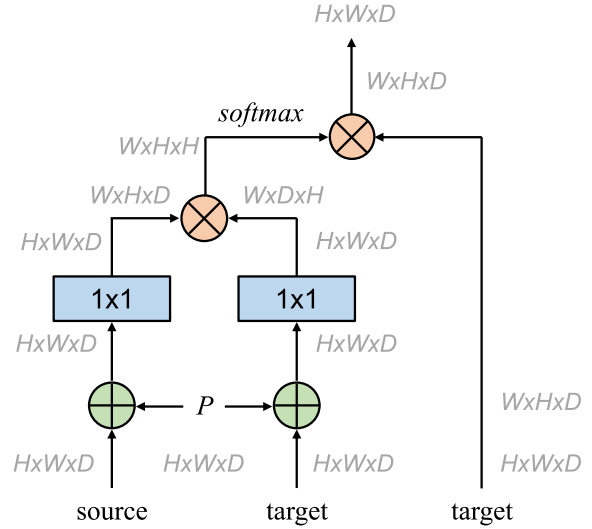

Figure 4: Computational graph of 1D cross attention in vertical direction. The positional encoding P is added to the input features. Two 1×1 convolutions are used to learn the attention weights. "$\oplus$" denotes element-wise addition and "$\otimes$" denotes matrix multiplication on the last two dimensions. The softmax operation is performed on the last dimension. The matrix dimensions are indicated with grey font, proper reshaping is performed when noted.

to find $f_{i,h,w}$ that can properly aggregate the column-wise information to aid in the subsequent 1D correspondence search (Fig. 1). Inspired by the success of Transformers [38] in modeling long-range dependencies, we propose to learn the combination coefficients with the attention mechanism.

However, different from the original attention that computes all pair-wise similarities, we only perform 1D attention in an axis-aligned manner. As illustrated in Fig. 3, the goal of the attention operation is to propagate the information of the red point (blue point's correspondence) onto the green one, which lies on the same row of the blue point. To achieve this, one naïve solution is applying a 1D cross attention operation between the same column of the source and target, to make the target feature aggregation dependent on the source feature. However, the same column in the source image may not contain the corresponding pixel (blue point), rendering it difficult to learn proper aggregation. To resolve this issue, we first perform a 1D self attention operation in the horizontal direction of the source feature before computing the cross attention, which propagates the information of the corresponding point on source image to the entire row.

The computational graph of 1D vertical cross attention is illustrated in Fig. 4. The inputs are self-attended source feature and original target feature. We also introduce position information $P \in \mathbb{R}^{H \times W \times D}$ in the attention computation where P is a fixed 2D positional encoding same as DETR [5]. We first use two 1×1 convolutions to project

$\boldsymbol{F}_1 + \boldsymbol{P}$ and $\boldsymbol{F}_2 + \boldsymbol{P}$ into the embedding space $\tilde{\boldsymbol{F}}_1$ and $\tilde{\boldsymbol{F}}_2$. Then the attention matrix on the vertical direction can be obtained by first reshaping $\tilde{\boldsymbol{F}}_1$ and $\tilde{\boldsymbol{F}}_2$ to $W \times H \times D$ and $W \times D \times H$, respectively, and then performing a matrix multiplication on the last two dimensions, which results in a $W \times H \times H$ matrix. By normalizing the last dimension with the softmax function, we obtain the attention matrix $\boldsymbol{A} \in \mathbb{R}^{W \times H \times H}$. The final attended feature $\hat{\boldsymbol{F}}_2$ can be computed by multiplying matrix $\boldsymbol{A}$ with the reshaped target feature ($W \times H \times D$) on the last two dimensions. By reshaping the resulting $W \times H \times D$ matrix to $H \times W \times D$, we obtain the final feature $\hat{\boldsymbol{F}}_2$. After the 1D cross attention operation, each position in $\hat{\boldsymbol{F}}_2$ has encoded the information of positions that are in the same column. This process can be similarly applied to 1D horizontal self attention computation by replacing the target features with source features and performing matrix multiplication on the width dimension.

3.2. 1D Correlation

With vertically aggregated feature $\hat{\boldsymbol{F}}_2$, we can perform a simple 1D search along the horizontal direction to construct a 3D cost volume $\boldsymbol{C} \in \mathbb{R}^{H \times W \times (2R+1)}$, similar to stereo methods [25, 41]. We use R to represent the search radius along the horizontal direction, and we have

$$\boldsymbol{C}(h, w, R+r) = \frac{1}{\sqrt{D}} \boldsymbol{F}_1(h, w) \cdot \hat{\boldsymbol{F}}_2(h, w+r) \quad (2)$$

where $\cdot$ denotes the dot-product operator, $r \in \{-R, -R+1, \cdots, 0, \cdots, R-1, R\}$, and $\frac{1}{\sqrt{D}}$ is a normalization factor to avoid large values after dot product following [38]. In our implementation, we pre-compute a 3D cost volume of size $H \times W \times W$ by performing matrix multiplication between $\boldsymbol{F}_1$ and $\hat{\boldsymbol{F}}_2$ on the width dimension. Then the equivalent form of Eq. (2) can be obtained by performing a lookup operation on the 3D cost volume with search radius R.

Although we only perform 1D correlations, our method can model 2D correspondence thanks to the attention operation. Specifically, the theoretical search range of our cost volume construction method is $(2R+1)(H+W) - (2R+1)^2$. This can be proven by substituting Eq. (1) into Eq. (2):

$$\boldsymbol{C}(h, w, R+r) = \frac{1}{\sqrt{D}} \boldsymbol{F}_1(h, w) \cdot \sum_{i=0}^{H-1} f_{i,h,w+r} \boldsymbol{F}_2(i, w+r)$$
$$= \frac{1}{\sqrt{D}} \sum_{i=0}^{H-1} f_{i,h,w+r} [\boldsymbol{F}_1(h, w) \cdot \boldsymbol{F}_2(i, w+r)],$$
$$(3)$$

where $f_{i,h,w+r}$ is defined with the 1D attention operation in Sec. 3.1. From term $\boldsymbol{F}_1(h, w) \cdot \boldsymbol{F}_2(i, w+r)$ in Eq. (3) with $i = 0, 1, \cdots H-1$, and $r = -R, -R+1, \cdots, 0, \cdots, R-1, R$, we can see that the search range of position (h, w)

spans vertically to the image height and horizontally to the maximum search radius R. Therefore, the theoretical search range for vertical attention and horizontal correlation is $H(2R+1)$. Similarly, when performing horizontal attention and vertical correlation, we can obtain another cost volume $\tilde{\boldsymbol{C}} \in \mathbb{R}^{H \times W \times (2R+1)}$, where

$$\tilde{\boldsymbol{C}}(h, w, R+r) = \frac{1}{\sqrt{D}} \sum_{j=0}^{W-1} f_{j,h+r,w} [\boldsymbol{F}_1(h, w) \cdot \boldsymbol{F}_2(h+r, j)].$$
$$(4)$$

In this case, the theoretical search range is $W(2R+1)$. Concatenating these two cost volumes results in a cost volume of shape $H \times W \times 2(2R+1)$, and the theoretical search range becomes $(2R+1)(H+W) - (2R+1)^2$, where $(2R+1)^2$ is the overlapping area of these two cost volumes.

As a comparison, previous local window based approaches [10, 18, 34] usually construct a cost volume of shape $H \times W \times (2R+1)^2$ and the search range is $(2R+1)^2$. Thus, our method enjoys a larger theoretical search range while maintaining a much smaller cost volume.

4. Flow Regression Framework

We verify the effectiveness of our proposed cost volume construction method with RAFT framework [37]. As illustrated in Fig. 2, we first extract $8\times$ downsampled features for source and target images, and then construct two 3D cost volumes with our proposed method in Sec. 3. Next, a shared update operator is applied iteratively to estimate the flow updates. At each iteration, a set of correlation values will be generated by looking up the 3D cost volume with current flow estimate. A context feature is also extracted from the source image with an additional network following RAFT, which is omitted in Fig. 2 for brevity. The correlations, together with the estimated flow and context feature, are then fed to a ConvGRU unit to produce a flow update, which is added to the current flow estimate.

In our framework, the lookup operation is defined by indexing the pre-computed cost volume with the current flow estimate. Specifically, if the current flow estimate is $\boldsymbol{f} = (f_x, f_y)$, then the lookup center becomes $(h + f_y, w + f_x)$ for pixel position (h, w). We index the 3D cost volumes both horizontally and vertically within a search range of R and obtain two 3D cost volumes ($H \times W \times (2R+1)$) that are concatenated for optical flow regression. We note that the correlation lookup is equivalent to Eq. (2) by modifying the correlation center to $(h + f_y, w + f_x)$.

5. Experiments

Datasets and evaluation setup. we consider two evaluation setups following previous methods [18, 34, 37]. First, we pre-train our model on FlyingChairs [10] and FlyingThings3D [25] datasets, and then evaluate the cross-dataset

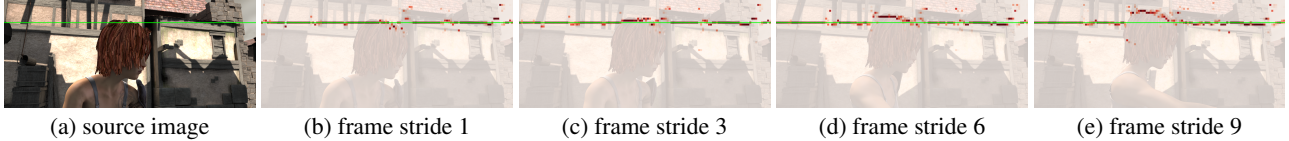| (a) source image | (b) frame stride 1 | (c) frame stride 3 | (d) frame stride 6 | (e) frame stride 9 |

Figure 5: Visualization of 1D *vertical* attentions. We fix the source image (5a) and select target frames (5b-5e) from a video sequence with gradually increased frame strides. The attention weights for the green row pixels are visualized onto the image, which shows the attentions generally focus on relevant pixels (*e.g.*, the moving head).

| (a) frame stride 1 | (b) frame stride 3 | (c) frame stride 6 | (d) frame stride 9 |

Figure 6: Visualization of 1D *horizontal* attentions. Similar phenomenon can be observed as Fig. 5.

Cost volume	Sintel (train, clean)		
	EPE	EPE (x)	EPE (y)
y attn, x corr	3.10	1.66	2.12
x attn, y corr	4.05	3.55	1.13
concat both	**1.98**	**1.48**	**0.94**

Table 1: Analysis on horizontal (x) and vertical (y) cost volumes. EPE (x) and EPE (y) represent the end-point-error of the horizontal and vertical flow component, respectively.

generalization ability on Sintel [4] and KITTI [26] training sets. Second, we perform additional fine-tuning on Sintel and KITTI training sets and then evaluate on the online benchmarks. The end-point-error (EPE) is reported in evaluation. For KITTI, another evaluation metric, F1-all, which denotes percentage of outliers for all pixels, is also reported. For ablation study, we also use the EPE in different motion magnitudes to better understand the performance gains. Specifically, we use s_{0-10}, s_{10-40} and s_{40+} to denote the EPE over regions with speed in $0 - 10$, $10 - 40$ and more than 40 pixels. For experiments on very high-resolution images, we mix FlyingThings3D, Sintel, HD1K [21] and Slow Flow [19] datasets for additional fine-tuning.

Implementation details. We implement our framework in PyTorch [27] and use AdamW [23] as the optimizer. We follow RAFT [37] for dataset schedule and training hyperparamters. Training is first conducted on FlyingChairs dataset for 100K iterations, followed by another 100K iterations on FlyingThings3D dataset. We also perform dataset-specific fine-tuning on Sintel and KITTI datasets. For training, we use 12 iterations for flow regression. For evaluation,

the iteration numbers for Sintel and KITTI are set to 32 and 24, respectively. The search range R in the cost volume lookup is set to 32, which correspondences to 256 pixels in the original image resolution. More implementation details are presented in *supplementary materials*.

5.1. Analysis

Ablation study. We first analyze the effectiveness of key components in our proposed method in Tab. 2. When not using 1D self and cross attentions, our cost volume degrades to pure vertical and horizontal 1D searches, which apparently loses too much information [1]. As a result, the performance drops a lot. We also note that from the detailed metrics in different motion magnitudes, our full model shows significant improvement on large motion (s_{40+}), demonstrating the effectiveness of the proposed method for large displacements. We also evaluate the effectiveness of self attention when computing the cross attention weights. By performing 1D self attention, 2D pixel relations are modeled more accurately, leading to better results. Meanwhile, the positional encoding used in the attention computation is helpful, as also demonstrated by previous works [38, 5].

We also analyze the role of each 3D cost volume plays in Tab. 1. We observe that the performance of horizontal or vertical flow is coupled with the correlation direction. Horizontal cost volume is mainly responsible for the horizontal flow estimation, and similarly for the vertical cost volume. Concatenating these two cost volumes gives the network necessary information for estimating both horizontal and vertical flow components.

Attention visualization. To better understand how our proposed method works, we further visualize the learned 1D cross attentions. For 1D vertical attention, the learned

Method	Sintel (train, clean)				Sintel (train, final)				KITTI (train)				
	EPE	s_{0-10}	s_{10-40}	s_{40+}	EPE	s_{0-10}	s_{10-40}	s_{40+}	EPE	s_{0-10}	s_{10-40}	s_{40+}	F1-all
w/o self & cross	2.94	1.75	4.65	24.96	4.72	2.43	6.69	30.27	14.01	0.99	3.87	29.06	38.25
w/o self	2.15	**1.44**	3.83	20.87	3.54	2.16	5.49	25.29	8.67	0.89	2.49	17.82	27.25
w/o pos	**1.98**	1.49	**3.80**	19.65	3.61	2.49	5.61	25.45	7.71	0.84	2.31	15.49	25.37
Flow1D	**1.98**	1.68	3.92	**18.86**	**3.27**	**2.06**	**5.18**	**23.76**	**6.69**	**0.80**	**2.18**	**13.41**	**22.95**

Table 2: Ablation study for our cost volume design. Models are trained on FlyingChairs and FlyingThings3D.

Method	Sintel (train)		KITTI (train)		Params	448×1024		1088×1920	
	Clean	Final	EPE	F1-all		Memory	Time (ms)	Memory	Time (ms)
RAFT [37]	1.43	2.71	5.04	17.40	5.26M	0.48GB	94	8.33GB	393
FlowNet2 [18]	2.02	3.14	10.06	30.37	162.52M	1.31GB	186	3.61GB	496
PWC-Net [34]	2.55	3.93	10.35	33.67	9.37M	0.86GB	24	1.57GB	84
Flow1D	1.98	3.27	6.69	22.95	5.73M	0.34GB	79	1.42GB	332

Table 3: Evaluation after training on FlyingChairs and FlyingThings3D datasets. Memory and inference time are measured for 448×1024 and 1088×1920 resolutions on a V100 GPU, and the iteration numbers are 12 for RAFT and our method.
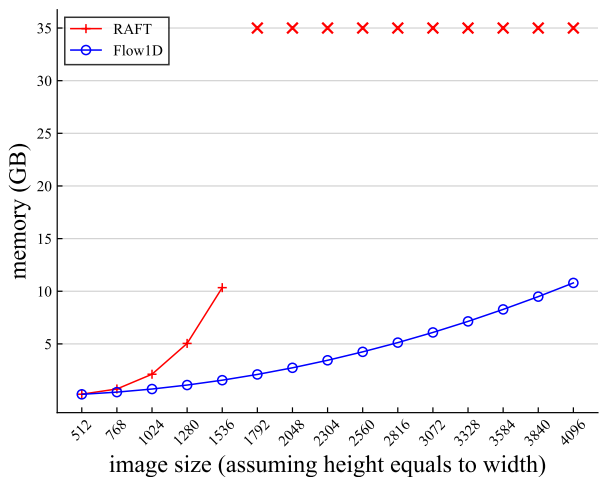

Figure 7: Memory consumption *vs.* input resolutions for RAFT and our method. The red cross $\times$ denotes that RAFT causes an out-of-memory error for high-resolution inputs even on a 32GB GPU.

attention map is of shape $H \times W \times H$, which includes the vertical attention weights for each pixel. we take of a row of pixels and visualize their vertical attentions in Fig. 5, and similarly for the horizontal attentions in Fig. 6. To investigate how attention maps change over time, we select target frames at different frame strides for comparison. It is observed that the learned attentions are quite sparse and generally focus on the most relevant pixels, verifying the working mechanism of our proposed method.

5.2. Comparison with Existing Cost Volumes

To demonstrate the superiority of our proposed cost volume construction method, we conduct comprehensive comparisons with existing cost volumes from different aspects.

Setup. We mainly compare with three representative cost volume construction methods: FlowNet2 [18]'s single scale cost volume, with stacked networks for refinement; PWC-Net [34]'s multiple small cost volumes with a coarse-to-fine framework and RAFT [37]'s 4D cost volume to estimate optical flow iteratively. The performance is evaluated after training on FlyingChairs and FlyingThings3D datasets.

Sintel and KITTI results. Table 3 shows the comprehensive evaluation results. In terms of accuracy, our method is higher than FlowNet2 and PWC-Net, especially on KITTI dataset, but is inferior to RAFT. Exhaustively constructing a large 4D cost volume is indeed advantageous to obtain highly accurate flow in RAFT, while our 3D cost volume may have the risk of missing relevant pixels. This phenomenon can be partially observed in the visualization of the learned attention maps in Fig. 5 and Fig. 6: although most pixels are able to find the right correspondences, there are few noises in the attention maps that cause missing pixels. It's possible that a better design of the attention matrix computation will further improve the performance. However, our method shows higher efficiency than RAFT in terms of both memory consumption and inference speed. The superiority will be more significant for high-resolution images. For example, at 1088×1920 image resolution, our method consumes $6\times$ less memory than RAFT[1].

High-resolution results. We further show some visual comparisons on the high-resolution (1080×1920) DAVIS [28] dataset in Fig. 8. We can achieve comparable results with RAFT while consuming $6\times$ less memory. We also

---

[1]Note although RAFT can reduce the memory by re-constructing cost volume at each iteration with a customized CUDA implementation, the inference time increases substantially in practice ($\sim 4\times$ slower)

Method	Sintel (train)		Sintel (test)		KITTI (F1-all)	
	Clean	Final	Clean	Final	(train)	(test)
FlowNet2 [18]	(1.45)	(2.01)	4.16	5.74	(6.8)	11.48
PWC-Net+ [35]	(1.71)	(2.34)	3.45	4.60	(5.3)	7.72
HD^3 [45]	(1.87)	(1.17)	4.79	4.67	(4.1)	6.55
VCN [43]	(1.66)	(2.24)	2.81	4.40	(4.1)	6.30
MaskFlowNet [47]	-	-	2.52	4.17	-	6.10
RAFT [37]	(0.77)	(1.27)	1.61	2.86	(1.5)	5.10
Flow1D	(0.84)	(1.25)	2.24	3.81	(1.6)	6.27

Table 4: Benchmark performance on Sintel and KITTI datasets. The numbers in the parenthesis are the results on the data that the methods have been fine-tuned on.

show additional results on 4K (2160 × 3840) resolution images in Fig. 9, which can not be processed by RAFT due to the huge memory consumption. We achieve satisfactory optical flow estimation while consuming only 5.8GB memory. More results can be found in *supplementary materials*.

Scalability. We further compare the scalability of our method with RAFT. For input feature map of size $H \times W \times D$, the computational complexity of constructing a 4D cost volume in RAFT [37] is $\mathcal{O}((HW)^2 D)$. As a comparison, ours is $\mathcal{O}(HW(H + W)D)$ for two 3D cost volumes. We measure the practical memory consumption under different input resolutions in Fig. 7, our method is able to scale to more than 8K resolution (4320 × 7680, memory consumption is 21.81GB) images while RAFT quickly causes an out-of-memory issue even on a high-end GPU that has 32GB memory, demonstrating the superiority of our method.

5.3. Benchmark Results

Sintel. For submission to Sintel dataset, we fine-tune on mixed KITTI [11, 26], HD1K [21], FlyingThings3D [25] and Sintel [4] datasets for 100K iterations. The evaluation results are shown in Tab. 4. Our method ranks second only to RAFT, outperforming previous representative methods such as PWC-Net [34] and FlowNet2 [18].

KITTI. We further fine-tune on the KITTI 2015 training set for 50K iterations. Table 4 shows the evaluation results. Our method performs better than PWC-Net+, while slightly inferior to MaskFlowNet, which is likely caused by the limited training data of KITTI dataset.

6. Conclusion

We have presented a new cost volume construction method for high-resolution optical flow estimation. By factorizing the 2D optical flow with 1D attention and 1D correlation, we are able to scale to more than 8K resolution images while maintaining competitive performance. We hope our new perspective can stimulate future research on cost

Figure 8: Comparisons on high-resolution (1080 × 1920) images from DAVIS dataset. We achieve comparable results with RAFT while consuming 6× less memory.

Figure 9: Optical flow prediction results on 4K resolution (2160 × 3840) images from DAVIS dataset.

volume compression and efficient high-resolution optical flow estimation.

References

[1] Nicola Ancona and Tomaso Poggio. Optical flow from 1-d correlation: Application to a simple time-to-crash detector. *International Journal of Computer Vision*, 14(2):131–146, 1995. 6

[2] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. High accuracy optical flow estimation based on a theory for warping. In *European Conference on Computer Vision*, pages 25–36. Springer, 2004. 2

[3] Thomas Brox and Jitendra Malik. Large displacement optical flow: descriptor matching in variational motion estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(3):500–513, 2010. 1, 2

[4] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European Conference on Computer Vision*, pages 611–625. Springer, 2012. 2, 6, 8

[5] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. *arXiv preprint arXiv:2005.12872*, 2020. 4, 6

[6] Qifeng Chen and Vladlen Koltun. Full flow: Optical flow estimation by global optimization over regular grids. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4706–4714, 2016. 2

[7] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating long sequences with sparse transformers. *arXiv preprint arXiv:1904.10509*, 2019. 3

[8] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014. 3

[9] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*, 2019. 3

[10] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2758–2766, 2015. 1, 2, 3, 5

[11] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361. IEEE, 2012. 8

[12] Heiko Hirschmuller. Stereo processing by semiglobal matching and mutual information. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):328–341, 2007. 2

[13] Berthold KP Horn and Brian G Schunck. Determining optical flow. In *Techniques and Applications of Image Understanding*, volume 281, pages 319–331. International Society for Optics and Photonics, 1981. 1, 2

[14] Asmaa Hosni, Christoph Rhemann, Michael Bleyer, Carsten Rother, and Margrit Gelautz. Fast cost-volume filtering for visual correspondence and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(2):504–511, 2012. 1, 2

[15] Zilong Huang, Xinggang Wang, Lichao Huang, Chang Huang, Yunchao Wei, and Wenyu Liu. Ccnet: Criss-cross attention for semantic segmentation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 603–612, 2019. 3

[16] Tak-Wai Hui, Xiaoou Tang, and Chen Change 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. 3

[17] Junhwa Hur and Stefan Roth. Iterative residual refinement for joint optical flow and occlusion estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5754–5763, 2019. 2, 3

[18] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas 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. 1, 2, 3, 5, 7, 8

[19] Joel Janai, Fatma Guney, Jonas Wulff, Michael J Black, and Andreas Geiger. Slow flow: Exploiting high-speed cameras for accurate and diverse optical flow reference data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3597–3607, 2017. 6

[20] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super slomo: 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. 1

[21] Daniel Kondermann, Rahul Nair, Katrin Honauer, Karsten Krispin, Jonas Andrulis, Alexander Brock, Burkhard Gussefeld, Mohsen Rahimimoghaddam, Sabine Hofmann, Claus Brenner, et al. The hci benchmark suite: Stereo and flow ground truth with uncertainties for urban autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 19–28, 2016. 6, 8

[22] Zhengqi Li, Tali Dekel, Forrester Cole, Richard Tucker, Noah Snavely, Ce Liu, and William T Freeman. Learning the depths of moving people by watching frozen people. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4521–4530, 2019. 1

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[24] Yao Lu, Jack Valmadre, Heng Wang, Juho Kannala, Mehrtash Harandi, and Philip Torr. Devon: Deformable volume network for learning optical flow. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2705–2713, 2020. 1, 2

[25] Nikolaus Mayer, Eddy Ilg, Philip Hausser, Philipp Fischer, Daniel Cremers, Alexey Dosovitskiy, and Thomas Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4040–4048, 2016. 4, 5, 8

[26] Moritz Menze and Andreas Geiger. Object scene flow for autonomous vehicles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3061–3070, 2015. 2, 6, 8

[27] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8026–8037, 2019. 6

[28] Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alexander Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv:1704.00675*, 2017. 7

[29] Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui, and Cordelia Schmid. Epicflow: Edge-preserving interpolation of correspondences for optical flow. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1164–1172, 2015. 1, 2

[30] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002. 2

[31] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. In *Advances in Neural Information Processing Systems*, pages 568–576, 2014. 1

[32] Frank Steinbrücker, Thomas Pock, and Daniel Cremers. Large displacement optical flow computation without warping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1609–1614. IEEE, 2009. 1

[33] Deqing Sun, Stefan Roth, and Michael J Black. Secrets of optical flow estimation and their principles. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2432–2439. IEEE, 2010. 1, 2

[34] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan 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. 1, 2, 3, 5, 7, 8

[35] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Models matter, so does training: An empirical study of cnns for optical flow estimation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(6):1408–1423, 2019. 1, 8

[36] Yi Tay, Mostafa Dehghani, Dara Bahri, and Donald Metzler. Efficient transformers: A survey. *arXiv preprint arXiv:2009.06732*, 2020. 3

[37] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision*, pages 402–419. Springer, 2020. 1, 2, 3, 5, 6, 7, 8

[38] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008, 2017. 2, 3, 4, 5, 6

[39] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. In *European Conference on Computer Vision*, pages 108–126. Springer, 2020. 3

[40] Sinong Wang, Belinda Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*, 2020. 3

[41] Haofei Xu and Juyong Zhang. Aanet: Adaptive aggregation network for efficient stereo matching. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1959–1968, 2020. 4, 5

[42] Jia Xu, René Ranftl, and Vladlen Koltun. Accurate optical flow via direct cost volume processing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1289–1297, 2017. 2

[43] Gengshan Yang and Deva Ramanan. Volumetric correspondence networks for optical flow. In *Advances in Neural Information Processing Systems*, pages 794–805, 2019. 1, 2, 3, 8

[44] Jiaolong Yang and Hongdong Li. Dense, accurate optical flow estimation with piecewise parametric model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1019–1027, 2015. 1

[45] Zhichao Yin, Trevor Darrell, and Fisher Yu. Hierarchical discrete distribution decomposition for match density estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6044–6053, 2019. 3, 8

[46] Christopher Zach, Thomas Pock, and Horst Bischof. A duality based approach for realtime tv-l 1 optical flow. In *Joint Pattern Recognition Symposium*, pages 214–223. Springer, 2007. 1

[47] Shengyu Zhao, Yilun Sheng, Yue Dong, Eric I Chang, Yan Xu, et al. Maskflownet: Asymmetric feature matching with learnable occlusion mask. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6278–6287, 2020. 2, 3, 8