MILA: Multi-Task Learning from Videos via Efficient Inter-Frame Attention

Donghyun Kim[1], Tian Lan[2], Chuhang Zou[2], Ning Xu[2],
Bryan A. Plummer[1], Stan Sclaroff[1], Jayan Eledath[2], Gerard Medioni[2]
[1]Boston University, [2] Amazon
[1]{donhk, bplum, sclaroff}@bu.edu, [2]{tianlan, ninxu, zouchuha, eledathj, medioni}@amazon.com

Abstract

Prior work in multi-task learning has mainly focused on predictions on a single image. In this work, we present a new approach for multi-task learning from videos via efficient inter-frame local attention (MILA). Our approach contains a novel inter-frame attention module which allows learning of task-specific attention across frames. We embed the attention module in a "slow-fast" architecture, where the slow network runs on sparsely sampled keyframes and the fast shallow network runs on non-keyframes at a high frame rate. We also propose an effective adversarial learning strategy to encourage the slow and fast network to learn similar features to well align keyframes and non-keyframes. Our approach ensures low-latency multi-task learning while maintaining high quality predictions. MILA obatins competitive accuracy compared to state-of-the-art on two multi-task learning benchmarks while reducing the number of floating point operations (FLOPs) by up to 70%. In addition, our attention based feature propagation method (ILA) outperforms prior work in terms of task accuracy while also reducing up to 90% of FLOPs.

1. Introduction

Many computer vision applications, such as autonomous driving and indoor navigation, require multi-task predictions from video streams (*e.g.*, [5, 6, 7]). For example, a self-driving system needs semantic segmentation at each time frame to understand what entities are around the car, and depth estimation to determine how far away each entity is. This makes multi-task learning methods ideal since their shared representation can boost performance on each task while also being more computationally efficient.

In this paper, we focus on efficient multi-task learning for dense pixel-wise predictions (*e.g.* semantic segmentation and depth estimation) by leveraging a monocular video. Figure 1 compares the tradeoff between performance and computational burden for existing multi-task learning methods (blue) and our method (red). Recent multi-task learning approaches for dense predictions primarily use single-frame predictions [17, 26, 30, 33] and often require heavy task-specific layers (illustrated in Figure 2(a-b)), or a naive concatenation of the features from two consecutive video frames [6] (Figure 2(c)), which require massive amounts of floating point operations (FLOPs) to compute). To address this, we propose MILA – multi-task learning framework for videos that exploits temporal cues using inter-frame local attention (ILA) modules as shown in Figure 2(d). Existing attention modules only attend to features in the current (single) frame [14, 30], but ILA efficiently learns to attend and propagate features from previous frames. ILA is far more efficient than the expensive optical-flow based feature warping, which is widely used in previous work [24, 50]. In addition, the performance of optical flow warping based methods can be affected by the quality of estimated optical flow, which may fail on fast motion or occluded objects.

Our MILA architecture embeds the ILA feature propaga-

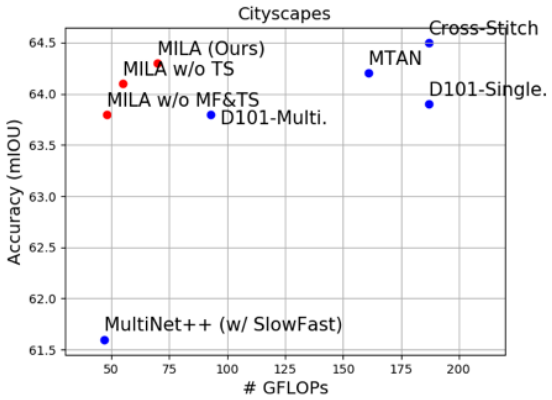

Figure 1: Comparison of the number of GFLOPs and mIOU performance of multi-task learning methods on Cityscapes. Our method (MILA) reduces computational burden significantly while maintaining accuracy. Our ILA module can be extended to attend task-specific (TS) and multi-frame features (MF) with minimal computations. We use the notation of each compared method and baseline from Sec. 4.2.

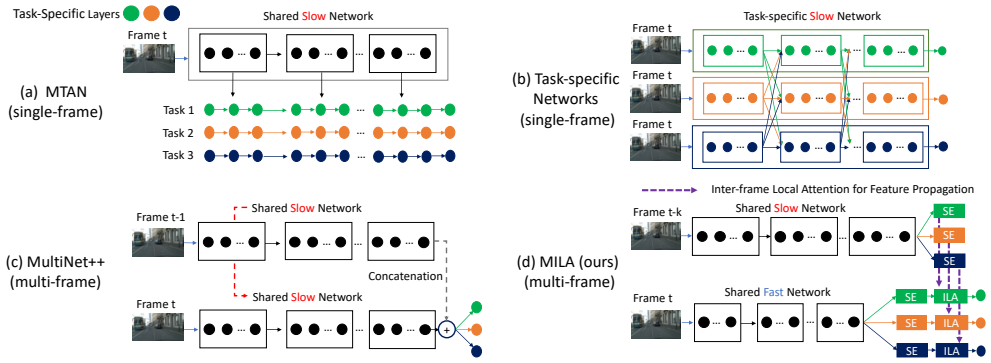

Figure 2: An illustration of difference between previous multi-task learning models and our multi-task model, MILA. Previous methods use only a $Slow$ network (*e.g.* ResNet101) for every frame ((a), (b), and (c)) and heavy task-specific layers for each task ((a) and (b)), which requires massive computations. In (d), we propose an efficient approach for multi-task learning from videos by utilizing a $Fast$ network (*e.g.* ResNet18) for non-keyframes and propagating the previous keyframe features from the $Slow$ network via our inter-frame local attention module (ILA). ILA is light-weight, accurate, and extended task-specific attention modules without requiring massive computations.

tion method in a *SlowFast* framework [13, 24], which can reduce computational cost while maintaining comparable accuracy. In the *SlowFast* architecture, keyframes are processed by a deep ($Slow$) network, and non-keyframes are processed by a shallow ($Fast$) network. Moreover, unlike the previous task-specific heavy layers, we show improvements in accuracy with our light-weight task-specific attention based ILA by leveraging temporal cues, and a novel adversarial learning strategy that encourages similar feature representations for both the $Slow$ and $Fast$ networks. Our ILA module differs from other attention-based approaches (*e.g.* [14]) in that we use attention to estimate the importance of features from different networks rather than the same network, which is a challenging problem that is not fully addressed by the existing attention modules. Figure 2 illustrates the difference between our approach (MILA) and existing multi-task learning methods.

We evaluate our approach on two standard multi-task learning benchmarks: Cityscapes [8] with outdoor scenes and NYUd v2 [36] with indoor scenes. As shown in Figure 1, MILA method achieves on-par accuracy compared to the state-of-the-art multi-task learning methods, while reducing the number of FLOPs by up to 70%. MILA reduces the computational burden by a large margin without compromising accuracy. Moreover, we show that the ILA module can be used as a standalone feature propagation method in videos: it is much faster compared to existing feature propagation methods, and more accurate than the state-of-the-art [24, 29] on semantic video segmentation.

Our contributions are:

- We address the task of video-based multi-task learning, which is not well explored in previous work. We present a multi-task learning framework via inter-frame local attention (MILA) that achieves competitive accuracy as compared to the state-of-the-art with a largely reduced computational cost.

- We introduce a new inter-frame local attention module (ILA) which learns task-specific features across frames. Our network is trained end-to-end with an adversarial loss for the *SlowFast* network.

- Our ILA module can be used as a standalone feature propagation method in video tasks such as semantic segmentation, achieving the top accuracy with up to 90% reduction of FLOPs.

2. Related Work

Multi-task learning (MTL) has shown improved accuracy or increased memory-efficiency for various tasks such as object classification, object detection and segmentation [2, 3, 20, 33, 38], joint scene geometry and semantic segmentation [6, 12, 26, 27, 30, 40, 46, 49]. Several methods are proposed to learn useful task-specific representations from shared representations or representations from other tasks [1, 33, 49, 48]. The cross-stitch unit [33] links representations between different tasks. Zhou *et al.* [49] propose a pattern-structure diffusion for propagating inter and intra task-specific representations. However, previous approaches on dense prediction tasks (*e.g.*, semantic segmentation) mainly focus on predictions from a single image. Chennupati *et al.* [6] learn from videos by concatenating the features from two consecutive frames. Contrary to prior work, we go beyond single-frame based prediction and learn from videos by aggregating and propagating features across multiple frames.

Although the shared representation of MTL can help improve generalization and reduce computational costs, it is also shown to potentially hurt accuracy due to the trade-off learning from multiple tasks [32, 37]. Kendall *et al*. [26] use homeostatic uncertainty to weight different tasks adaptively during training. Other methods introduce complex task-specific layers (*e.g*. task-specific backbone) [30, 33, 35] that also significantly increases the computational burden. In contrast, we show that our lightweight task-specific model design for our inter-frame attention module is able to achieve competitive task accuracy at a much lower computational cost via video learning.

Feature propagation has been widely used in video applications to exploit temporal cues across frames [6, 15, 24, 29, 34, 50] in order to reduce computational costs. Jain *et al*. [24] reduce the inference cost by combining the predictions of two network branches: a deep reference branch that computes detailed features from keyframes, and a shallower update branch that incorporates less detailed features at each frame with the wrapped features from a recently met keyframe. This is similar to the *SlowFast* [13] design for video recognition. However, optical flow based feature propagation [24] increases the computational cost with limited improvements in accuracy compared to the simple concatenation of features [6]. [25, 29] use spatially variant convolution layers (SVC) for feature propagation which is faster than optical flow warping. Our network, MILA, stems from the spirit of the *SlowFast* network, and we use our light-weight inter-frame local attention (ILA) module for feature propagation instead of the expensive optical flow based approach. We also perform dense feature propagation between every neighboring frame, in addition to the sparse propagation between keyframes and non-keyframes only.

Attention modules are widely used in various tasks such as natural language processing (NLP) [10, 41], semantic segmentation [14, 23, 28, 41, 47], image classification [22, 44, 42], and video object detection [19, 43, 45]. Channel-wise attention in CNNs has been proposed in [22, 42]. Vaswani *et al*. [41] propose a self-attention module for a translation task by extracting global dependencies from input sequences. The self-attention first computes feature representations for query, key, and value, then computes global attention weights by measuring the similarity between the query and key. The final value can be obtained by a weighted sum of values from the sequence of input. Some video object detection methods use global/local attention modules [19, 43, 45] for inter-frame features. These global/local spatial attention modules will not work well in our framework as we need to attend between two different representations (from $Slow$ and $Fast$ networks). In our work, we also add an adversarial loss in our training scheme (Sec 3.3) to facilitate the attention module for the *SlowFast* network and improve performance.

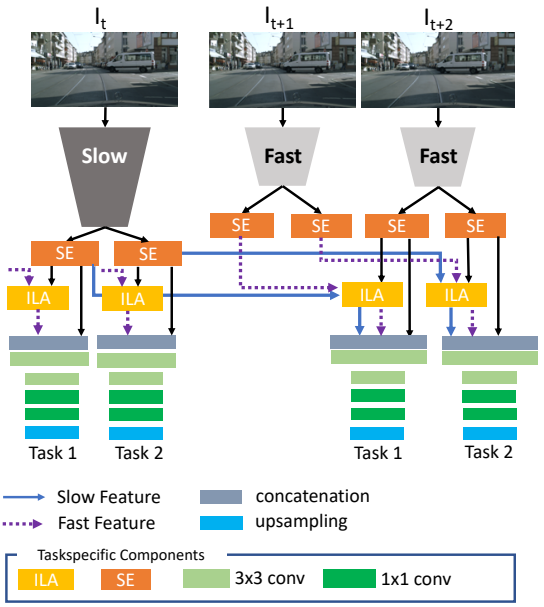

Figure 3: An illustration of our network architecture and the inference step of a keyframe I_t and a non-keyframe I_{t+2}. ILA propagates multi-frame features to the current frame.

3. Multi-task Learning via Inter-Frame Local Attention

We propose Multi-task Learning via Inter-Frame Local Attention (MILA), which is a computationally efficient multi-task learning model, with high quality prediction by leveraging temporal cues in video streams. Effectively learning spatial and temporal cues of different tasks in a light-weight and efficient manner is a challenge in real-time video applications. Inspired by the *SlowFast* network [13, 24], we build an efficient multi-task network with a two-branch design: the $Slow$ branch runs on sparsely sampled keyframes and the light-weight $Fast$ network runs on non-keyframes. Unlike previous work that relies on heavy task-specific layers [30, 33], we introduce a new light-weight task-specific attention module to learn and propagate task-specific features across frames. Figure 3 shows the architecture of the proposed network.

In the following, we first explain our multi-task network architecture (MILA) in Sec. 3.1. Then, we introduce our novel task-specific inter-frame local attention (ILA) module in Sec. 3.2, and an adversarial loss that further boosts the overall performance in Sec. 3.3.

3.1. Architecture Overview

MILA consists of two components: 1) a shared encoder network: a $Slow$ network that operates on sparsely sampled keyframes; a $Fast$ network runs on other frames. 2) M task-specific decoder networks, one for each task. Each decoder network learns to attend to task-specific features

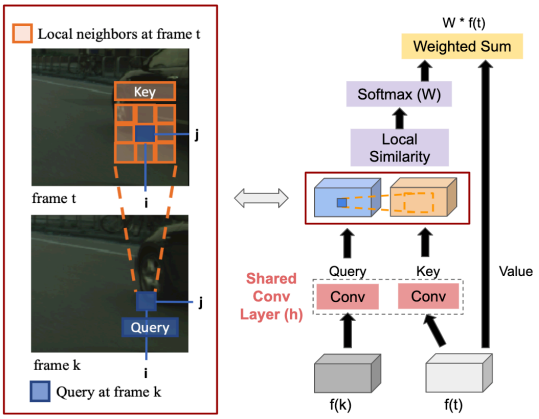

Figure 4: Inter-frame local attention (ILA) accounts for motion by finding local attention weights in inter-frames. With the shared conv layer, our module generates high attention weighs on the similar features between frames.

from previous frames and the current frame (in Figure 3).

The input is a sequence of N RGB frames $I = \{I_1, I_2, \ldots, I_N\}$ from a monocular video, and the output is pixel-level predictions on M tasks, $Y = \{y_1, y_2, ..., y_M\}$. At each time step $t \in \{1, 2, \ldots, T\}$, we encode frame I_t using the $Slow$ network if it is a keyframe, and the $Fast$ network otherwise. In our implementation, we use ResNet-101 as the $Slow$ network and ResNet-18 as the $Fast$ network. The encoder is shared among all tasks. We use $Slow(I_t)$ to denote features encoded by the $Slow$ network and $Fast(I_t)$ for features encoded by the $Fast$ network.

At the decoder step, we perform predictions on each task with a task-specific decoder $\{D_1, D_2, \ldots, D_M\}$, where M is the total number of tasks. Each task-specific decoder consists of squeeze-excitation (SE) blocks on top of shared features from the encoder, inter-frame local attention (ILA) modules to extract and propagate task-specific features across frames and a set of conv layers. In order to fully leverage temporal information, we enable multi-frame feature propagation: a non-keyframe receives features propagated from the last keyframe and the last non-keyframe; a keyframe receives features propagated from the last non-keyframe. This is different from existing feature propagation [24, 29] which only propagates features from a keyframe to a non-keyframe.

3.2. Inter-Frame Local Attention (ILA)

The key challenge for attention based feature propagation is how to leverage inter-frame temporal cues to propagate features efficiently and effectively. We introduce a light-weight inter-frame local attention (ILA) module for feature propagation. As illustrated in Figure 4, ILA computes local attention weights W from the feature maps of two different frames (either two neighboring frames or a non-keyframe and a keyframe) to exploit local motion changes.

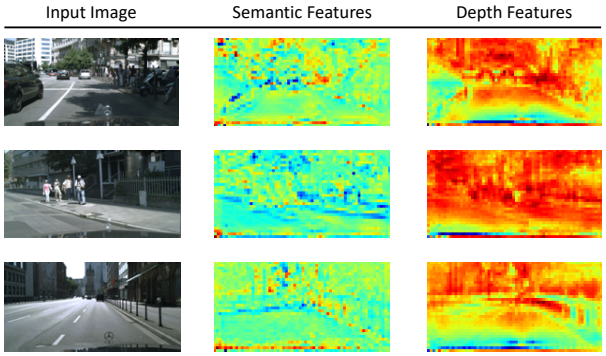

Figure 5: Visualization of task-specific features from our task-specific attention module.

Given a pair of frames I_t and I_k, ILA operates on feature maps f_t and f_k and propagates features from f_t to f_k. In our design (see Figure 3), the feature maps are the output of task-specific squeeze-and-excitation (SE) blocks [22]. For each pixel on the feature map f_k, we propagate the features from f_t based on a weighted combination of pixels in a local neighborhood.

$$f_{t\to k}(i,j) = \sum_{x=-L/2}^{L/2} \sum_{y=-L/2}^{L/2} W_{i,j}(x,y) f_t(i+x, j+y), \tag{1}$$

where (i, j) denotes the pixel location in the image, L is the window size and W is the attention weight obtained by measuring the similarity between the two feature maps f_t and f_k. The attention weight matrix W is defined in the following:

$$W_{i,j}(x,y) = \text{softmax}(h(f_k(i,j)) \cdot h(f_t(i+x, j+y))), \tag{2}$$

where $W_{i,j}(x, y)$ is the attention weight which measures the similarity between features at position (i, j) and $(i+x, i+y)$ of the two feature maps, respectively. h is a 3×3 convolution layer shared between the two feature maps to capture the semantic information in a local window around pixel (i, j). We use inner product to capture the similarities. Then a softmax layer is applied to ensure the sum of weights equals to 1. Note that ILA is performed only on local neighborhoods, resulting in reduced computational cost as compared to existing global attention modules [14].

3.2.1 Task-specific Attention

A common challenge in multi-task learning asks how to balance the shared and task-specific features. A heavily shared representation can reduce computational costs and

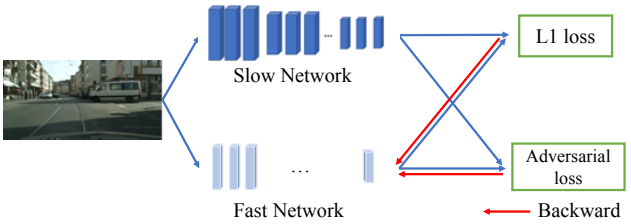

Figure 6: Adversarial learning. In order to let our attention module (ILA) capture temporal consistency, we adopt an adversarial learning strategy for training, where we use a combination of L1 and adversarial loss for the $Fast$ network to mimic the features learned by the $Slow$ network.

can help prevent over-fitting, but it can also hurt accuracy due to limited model capacity to handle multiple tasks [32]. To solve this issue, methods that add extra task specific layers to the multi-task network [30, 33, 35] gained popularity during the recent years and achieved higher task accuracy. The drawback is that the complex task-specific layers also significantly increase the computational burden.

Our ILA module is task-specific in order to learn discriminative task-specific features. In contrast to prior work, ILA learns to select and propagate features from previous frames rather than attending the features in the current frame. Leveraging temporal information drastically reduces the required complexity of task-specific layers as the model capacity and discriminative power are shared across multiple frames. Unlike the previous heavy attention modules in [14, 30], the other advantage is that ILA only attends to features from the task-specific SE blocks within a local window from previous frames. This assumption on temporal consistency reduces the computational cost of ILA. Compared to state-of-the-art attention-based multi-task network [30], MILA achieves better accuracy with 54% reduction of FLOPs.

Visualization of the learned task-specific features are shown in Figure 5. We can see clear differences in feature patterns for different tasks. Semantic segmentation features highlight object patches, lines and boundaries, while the depth features highlight foreground and background. This confirms the effectiveness of ILA as a feature selector to focus on parts that are discriminative for each task.

3.3. Boosting ILA for *SlowFast*

ILA assumes similar features propagate across frames. The high-level idea is similar to optical flow which assumes color constancy between pixels in consecutive images in order to capture motion. However, different backbones (e.g. ResNet-101 and ResNet-18) from the $Slow$ and $Fast$ branches cannot guarantee learning similar features for similar image patches. Thus, naive attention modules could not improve accuracy in our experiments (see Table 3).

We adopt adversarial learning to train the network so that the $Fast$ network learns similar features to the more accurate $Slow$ network. Figure 6 illustrates our approach. During training, we use a discriminator D [16, 18] to classify whether the features are output of the $Slow$ network or the $Fast$ network, and the $Fast$ network is trained to confuse the discriminator by "mimicking" the output features of the $Slow$ network. In practice, we observed combining $L1$ loss with the adversarial loss lead to improved accuracy. Our loss function $\mathcal{L}$ is defined in the following:

$$\begin{aligned} \mathcal{L} &= \min(\alpha\mathcal{L}_{L1} - \beta \min_{D} \mathcal{L}_{adversarial}) \\ \mathcal{L}_{L1} &= |Slow(I_t) - Fast(I_t)| \\ \mathcal{L}_{adversarial} &= \log D(Slow(I_t)) + \log(1 - D(Fast(I_t))), \end{aligned} \tag{3}$$

where $Slow(I_t)$ and $Fast(I_t)$ are the features of the $Slow$ and $Fast$ backbones on image I_t. The loss function L enforces the $Fast$ network to mimic the features learned from the $Slow$ network. D is only used during training and does not increase computation in inference time.

4. Experiments

We validate our approach (MILA) in the following two aspects for both accuracy and computation cost. (1) Comparison with the state-of-the-art multi-task learning approaches on videos, the ablation study for our proposed task-specific ILA, and our training losses. (2) The efficacy of our attention based feature propagation approach (ILA) compared with other feature propagation methods.

4.1. Implementation Details

We implement MILA using PyTorch. We train MILA using ADAM optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.99$. The learning rate is $1e^{-4}$ and batch size is 4. The training loss converges after 50 epochs. For the adversarial loss in Eq. 3, we set $\alpha = \beta = 1$. ILA computes on a window size of $L = 5$ in Eq. 2. We use DeepLab-ResNet101 [4, 21, 31] as a $Slow$ network and DeepLab-ResNet18 as a $Fast$ Network. The backbones are pre-trained on ImageNet [9] and finetuned for multi-task learning. For brevity, Deeplab-ResNet101 is denoted as D101. D101-18 refers to the *SlowFast* network with DeepLab-ResNet101 and DeepLab-ResNet18. Each task-specific decoder consists of three convolution layers with kernel size of 3x3, 1x1, and 1x1 respectively, and feature size of 512 and 256 in between.

Keyframe Interval. We train our network with a fixed keyframe interval of $K = 5$ following [24] (every 5-th frame is a keyframe). For evaluation, since frames in a video are sparsely annotated (*e.g.* 20-th frame in a video clip) in existing datasets, we measure performances of an annotated frame by running our method for all possible

Model	Segmentation		Depth		Normal Estimation					GFLOPs
					Angle Dist. ↓		Angle° Within ↑			
	mIOU ↑	Acc. ↑	Abs. ↓	Rel. ↓	Mean	Median	11.25°	22.5°	30.0°	
D101-SingleTask	37.2	75.0	39.3	16.6	22.8	16.6	35.7	62.8	73.9	236
D101-MultiTask	37.1	**75.3**	39.0	16.3	23.7	17.6	34.0	60.2	71.7	79
MTAN-Seg. [30]	17.7	55.3	59.0	25.8	31.4	25.4	23.2	45.7	57.6	178
Cross-Stitch-Seg. [30]	14.7	50.2	64.8	28.7	33.6	28.6	20.1	40.5	52.0	213
MTAN*	37.1	74.3	40.0	16.9	23.9	18.1	33.5	59.5	70.4	151
Cross-Stitch*	37.5	74.5	39.5	16.2	**22.7**	**16.5**	**36.8**	**63.0**	**73.8**	236
MultiNet++* [6]	32.8	73.1	41.1	17.3	24.4	18.1	33.4	58.9	70.2	**40**
MILA w/o MF&TS	36.6	74.8	39.2	16.5	23.5	17.4	34.2	60.4	71.8	41
MILS w/o TS	37.0	75.0	38.9	16.4	23.3	17.2	34.9	60.9	72.2	46
MILA (Ours)	**38.1**	75.1	**38.6**	**16.1**	23.2	17.0	35.4	61.8	72.5	70

Table 1: Comparisons for video based Multi-task learning on NYUd v2 dataset. * means training with the same Deeplab-ResNet101 backbone as ours. Cross-stitch* shows better results in the normal estimation task mostly because it contains task-specific backbones. Our method obtains high-quality predictions while reducing the massive computational burden.

keyframe interval offsets $[0, K-1]$ and report the averaged accuracy and GFLOPs. For the evaluation of ILA on semantic video segmentation, we use the same keyframe intervals as the compared methods (5 and 10).

4.2. Setup

Datasets. We evaluate our MILA on two widely used public video datasets: Cityscapes [8] and NYUd v2 [36]. We follow the evaluation protocols as in Liu *et al*. [30]: on Cityscapes, we perform 2 *task predictions* including 7-class segmentation and depth estimation, where images are resized to 256×512 to boost up training process; on the NYUd v2 dataset, we perform 3 *task predictions* including 13-class segmentation, depth estimation, and normal estimation, with input images resized to 288×384. For evaluation of ILA on the single task of semantic video segmentation, we perform 19-class segmentation the same as the state-of-the-art by Jain *et al*. [24].

Metrics. For semantic segmentation, we use mean intersection-over union (mIoU) metric and pixel accuracy (PA). For depth estimation, we evaluate on absolute and relative depth errors from the ground truth. For normal estimation, we measure the mean and median angle distances between the predicted angles and ground-truth angles. We also measure the percentage of pixels that are within the angles of $11^\circ, 22.5^\circ, 30^\circ$ to the ground-truth. We compare on computation cost based on GFLOPs following [13, 39, 51] and use the "thop" library [1] for counting GLFOPs.

Baselines. We compare with state-of-the-art multi-task learning methods: **MTAN** [30], **Cross-Stitch** network [33] and **MultiNet++** [6]. MTAN and Cross-Stitch are single frame based, while MultiNet++ uses multi-frame inputs. Although MTAN and Cross-Stitch used different backbones in their respective papers, for a fair comparison, we report the performance of the two using the same DeepLab-ResNet101 backbone as our *Slow* network (which shows better performance than the backbones from their papers). We use the four outputs of each group of layers containing the residual blocks in the backbone as input for the attention modules for the two methods (see Figure 2). For our method, and MultiNet++, we use the *SlowFast* network with ResNet101 and ResNet18. We also compare with two other baselines: **D101-SingleTask**, which uses a separate ResNet101 backbone for training each task without any shared features; **D101-MultiTask**, which uses the shared ResNet101 backbone with task-specific decoders.

[1] https://github.com/Lyken17/pytorch-OpCounter

Model	Segmentation		Depth		GFLOPs
	mIOU ↑	Acc. ↑	Abs. ↓	Rel. ↓	
D101-SingleTask	63.9	94.4	**1.02**	25.3	187
D101-MultiTask	63.8	94.4	1.06	31.9	93
MTAN-SegNet [30]	53.0	91.1	1.44	33.6	168
MTAN *	64.2	94.5	1.06	26.3	161
Cross-Stitch*	**64.5**	94.5	1.04	33.0	187
MultiNet++ [6]	61.6	93.9	1.08	28.5	**47**
MILA w/o MF&TS	63.8	94.4	1.05	32.9	48
MILA w/o TS	64.1	94.5	1.03	31.5	55
MILA (Ours)	64.3	**94.6**	**1.02**	**25.2**	70

Table 2: Comparison for video based multi-task learning on the Cityscapes dataset. * means training with the same Deeplab-ResNet101 backbone. Ours and MultiNet++ use the D101-18 backbone.

4.3. Video Based Multi-Task Learning

Quantitative results. We report the performance of video based multi-task learning on the NYUd v2 dataset in Table 1 and the Cityscapes dataset in Table 2 respectively. In Table 1, on the NYUd v2 dataset, we outperform other approaches for depth estimation and one metric of mIOU for semantic segmentation, with ranked 2nd segmentation accuracy. MILA shows slightly worse performance

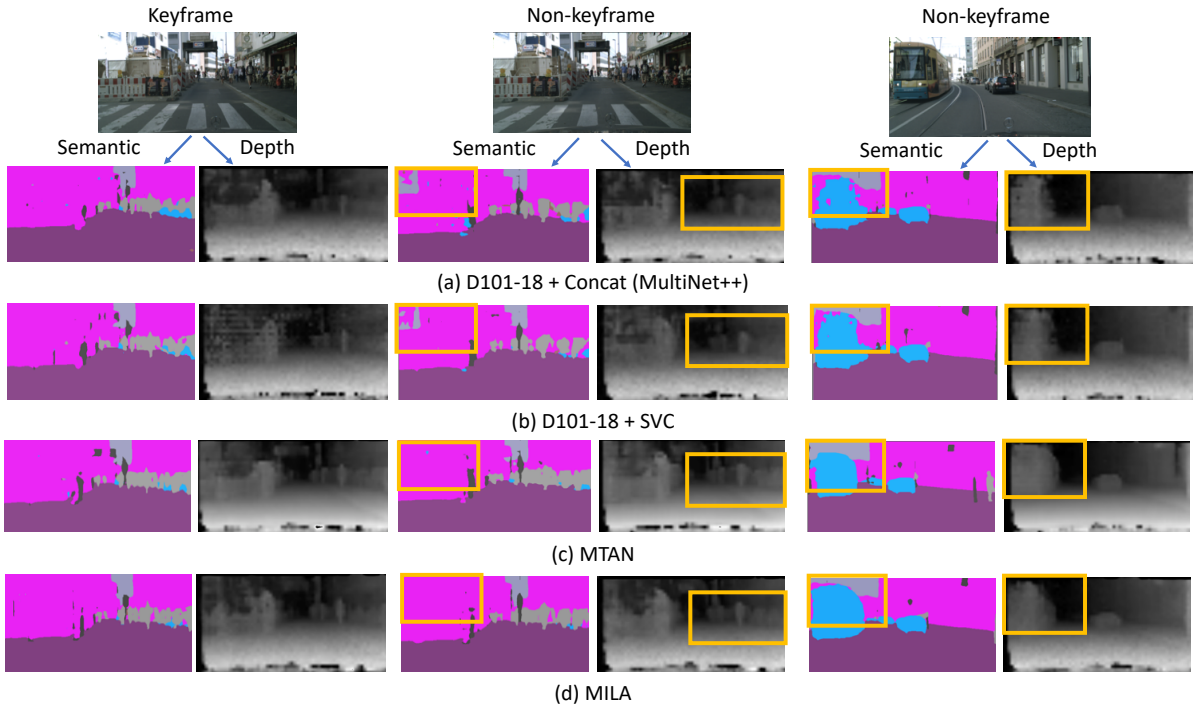

Figure 7: Qualitative results on Cityscapes. We choose a keyframe and non-keyframe with offset 4. For the keyframe, the three methods produce very similar qualitative results. For non-keyframes, the baseline methods (a, b) performs worse (see orange boxes) but our method (d) still obtains robust performance even compared with the $Slow$-only network (c).

for normal estimation than Cross-Stitch. This is because Cross-Stitch has task-specific backbones, while MILA use a single shared one, which is significantly computationally efficient, saving 70% of computational costs (236 vs. 70 GFLOPs) compared to Cross-Stitch [33]. MILA also saves 46% (70/151 GFLOPs) of computations compared to MTAN [30] as shown in Table 1 by replacing task-specific heavy layers with light-weight task-specific ILAs. We rank the 2nd for GFLOPs right after MultiNet++ but with much better accuracy. In Table 2, MILA outperforms all other methods for depth estimation and achieves the the best accuracy for semantic segmentation, while ranking the 2nd for mIOU on Cityscapes.

Qualitative results. We show in Figure 7 sample qualitative results from the Cityscapes dataset. Baselines with a *SlowFast* model (a,b) decreases accuracy and outputs noisy predictions in non-keyframes compared to a keyframe. But our method (d) produces robust predictions on non-keyframes even compared with MTAN (c) which uses $Slow$-only network. Please check more results and analyses in our supplementary material.

Ablation study. The last three rows in Tables 1 and 2 show the ablation study of MILA method. **"MILA w/o TS"** means our approach without task-specific attention design. **"MILA w/o MF&TS"** means without both task-specific design and feature propagation for neighboring frames. **"MILA w/o MF&TS"** achieves similar performance as D101-MultiTask while saving 48% computations. MILA outperforms D101-MultiTask and reduces 25% of computations. MILA approach shows the best performed accuracy with a small increase in GFLOPs.

Backbone	L1	Adv.	ILA	mIOU (↑)	Depth Err. (↓)
D101-18				61.6	1.08
D101-18	✓			61.8	1.07
D101-18			✓	61.9	1.08
D101-18	✓		✓	63.3	**1.05**
D101-18	✓	✓	✓	**63.8**	**1.05**

Table 3: Ablation study of ILA on Cityscapes. ILA with L1 loss and adversarial loss (denoted by Adv.) leads to clear improvement. ILA without the proposed losses obtains similar performances as just simple concatenation [6]. These losses do not increase GFLOPs in inference time.

We show the ablation study on our inter-frame local attention (ILA). Table 3 reports the impact of the adversarial and L1 losses (Eq. 3). In the third line of Table 3, it shows that the local attention module alone does not greatly improve the performances. However, when ILA is combined with the proposed losses, it significantly increases the accuracy. In addition, we show the ablation study for the local window size in Table 4. ILA is not sensitive to small

Backbone	Window Size	mIOU (↑)	Depth Err. (↓)
D101-18	3x3	64.1	**1.02**
D101-18	5x5	**64.3**	**1.02**
D101-18	7x7	64.2	**1.02**
D101-18	15x15	64.0	1.04
D101-18	Global (64)	62.4	1.10

Table 4: Ablation study on the kernel size in ILA on Cityscapes. ILA performs better than global attention for feature propagation.

Backbone	K	Feature Prop.	mIOU (%)
D101-18	5	Optical flow	72.1
D101-18	5	ILA (Ours)	**73.2**
D101-18	10	Optical flow	69.8
D101-18	10	ILA (Ours)	**72.1**
D101-34	5	Optical flow	72.4
D101-34	5	ILA (Ours)	**74.3**
D101-34	10	Optical flow	70.1
D101-34	10	ILA (Ours)	**73.8**

Table 5: Comparison with optical-flow based feature propagation [24] for the semantic segmentation task on Cityscapes. A keyframe interval is denoted by K.

changes in the window size, but performance drops significantly when the window size is global. In addition, a small window size is faster as it attends to less number of neighbors than the large window size.

4.4. Detailed Analysis on Feature Propagation

We compare ILA with the two feature propagation methods:(1) optical flow based warping with FlowNet-S [11], which shows state-of-the-art performance on the single task of semantic video segmentation for the Accel method [24] and (2) spatially variant convolution layers (SVC) [29] which is proposed to propagate features from a keyframe to non-keyframes. We use the same backbone for all methods for a fair comparison.

Performance comparison. To compare with optical flow based warping [24], we follow their semantic segmentation evaluation protocol in Table 5. Feature propagated with ILA obtains higher accuracy than optical flow-based warping [24]. We observe that the accuracy improvements of ILA are more evident in the higher keyframe interval.

In Table 6, we provide a comparison with SVC [29] on Cityscapes and NYUd v2 on multi-task learning. ILA outperforms all other methods in the two datasets while requiring less computational burden. We observe that the quality of optical flow estimation is poor in this evaluation protocol (*i.e.* low-resolution images), which results in significantly worse performance.

Space and computation cost. In Table 7, we show the

	Feature Prop.	Segmentation		Depth	
		mIOU ↑	Acc. ↑	Abs. ↓	Rel. ↓
(a)	Cityescapes				
	SVC [29]	62.3	94.0	1.06	33.3
	ILA (Ours)	**63.8**	**94.4**	**1.05**	**32.9**
(b)	NYUv2				
	SVC [29]	35.7	74.7	40.3	17.0
	ILA (Ours)	**36.6**	**74.8**	**39.2**	**16.5**

Table 6: Comparison with feature propagation methods with D101-18 backbone on Cityscapes and NYUv2

Feature Propagation	GFLOPs	# Conv.	# Param
(a) Input size: 258×512			
Optical flow [11, 24]	7.5	23	38M
SVC [29]	5.4	3	3M
ILA (Ours)	**0.2**	**1**	**0.2M**
(b) Input size: 1024×2048			
Optical flow [11, 24]	71.2	23	38M
SVC [29]	108	3	3M
ILA (Ours)	**5.4**	**1**	**0.2M**

Table 7: Comparison on feature propagation modules. SVC represents the method of [29]. Our method is light-weight and computationally efficient.

comparison of GFLOPs, the number of convolutional layers and the number of parameters for the optical flow warping, SVC, and our ILA. We report numbers given different input sizes. ILA consists of only one convolutional layer, making it much more memory efficient than the other two methods. For GFLOPs, other methods require more computations than ILA and the gain is more evident when the input size is larger. ILA takes only 4% (0.2/5.4 GFLOPs) of computations in the SVC feature propagation [29]. Additional analyses can be found in our supplementary material.

5. Conclusion

We present an efficient and effective multi-task learning framework on video streams. We propose a novel task-specific inter-frame local attention (ILA) module, which accounts for motion and propagate discriminative task-specific features over time in a spatial-variant manner. Our attention module is much faster, more accurate, and modular compared to prior feature propagation methods. Our inter-frame local attention module can be used to extract task-specific features with minimal computation compared to existing heavy task-specific layers. Our experiments show that our method significantly reduces the computational cost without compromising accuracy compared to the state-of-the-art multi-task learning models.

References

[1] Felix JS Bragman, Ryutaro Tanno, Sebastien Ourselin, Daniel C Alexander, and Jorge Cardoso. Stochastic filter groups for multi-task cnns: Learning specialist and generalist convolution kernels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1385–1394, 2019. 2

[2] Kai Chen, Jiangmiao Pang, Jiaqi Wang, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jianping Shi, Wanli Ouyang, et al. Hybrid task cascade for instance segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4974–4983, 2019. 2

[3] Liang-Chieh Chen, Alexander Hermans, George Papandreou, Florian Schroff, Peng Wang, and Hartwig Adam. Masklab: Instance segmentation by refining object detection with semantic and direction features. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4013–4022, 2018. 2

[4] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017. 5

[5] Yaran Chen, Dongbin Zhao, Le Lv, and Qichao Zhang. Multi-task learning for dangerous object detection in autonomous driving. *Information Sciences*, 432:559–571, 2018. 1

[6] Sumanth Chennupati, Ganesh Sistu, Senthil Yogamani, and Samir A Rawashdeh. Multinet++: Multi-stream feature aggregation and geometric loss strategy for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 1, 2, 3, 6, 7

[7] Sauhaarda Chowdhuri, Tushar Pankaj, and Karl Zipser. Multinet: Multi-modal multi-task learning for autonomous driving. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1496–1504. IEEE, 2019. 1

[8] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016. 2, 6

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 5

[10] Yuntian Deng, Yoon Kim, Justin Chiu, Demi Guo, and Alexander Rush. Latent alignment and variational attention. In *Advances in Neural Information Processing Systems*, pages 9712–9724, 2018. 3

[11] 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. 8

[12] David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE international conference on computer vision*, pages 2650–2658, 2015. 2

[13] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6202–6211, 2019. 2, 3, 6

[14] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019. 1, 2, 3, 4, 5

[15] Raghudeep Gadde, Varun Jampani, and Peter V Gehler. Semantic video cnns through representation warping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4453–4462, 2017. 3

[16] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 5

[17] Yuan Gao, Jiayi Ma, Mingbo Zhao, Wei Liu, and Alan L Yuille. Nddr-cnn: Layerwise feature fusing in multi-task cnns by neural discriminative dimensionality reduction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3205–3214, 2019. 1

[18] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014. 5

[19] Chaoxu Guo, Bin Fan, Jie Gu, Qian Zhang, Shiming Xiang, Véronique Prinet, and Chunhong Pan. Progressive sparse local attention for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3909–3918, 2019. 3

[20] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 5

[22] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018. 3, 4

[23] 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

[24] Samvit Jain, Xin Wang, and Joseph E Gonzalez. Accel: A corrective fusion network for efficient semantic segmentation on video. In *Proceedings of the IEEE Conference*

on Computer Vision and Pattern Recognition, pages 8866–8875, 2019. 1, 2, 3, 4, 5, 6, 8

[25] Zhengkai Jiang, Peng Gao, Chaoxu Guo, Qian Zhang, Shiming Xiang, and Chunhong Pan. Video object detection with locally-weighted deformable neighbors. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 8529–8536, 2019. 3

[26] Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7482–7491, 2018. 1, 2, 3

[27] Iasonas Kokkinos. Ubernet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6129–6138, 2017. 2

[28] Yanwei Li, Xinze Chen, Zheng Zhu, Lingxi Xie, Guan Huang, Dalong Du, and Xingang Wang. Attention-guided unified network for panoptic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7026–7035, 2019. 3

[29] Yule Li, Jianping Shi, and Dahua Lin. Low-latency video semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5997–6005, 2018. 2, 3, 4, 8

[30] Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1871–1880, 2019. 1, 2, 3, 5, 6, 7

[31] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015. 5

[32] Kevis-Kokitsi Maninis, Ilija Radosavovic, and Iasonas Kokkinos. Attentive single-tasking of multiple tasks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1851–1860, 2019. 3, 5

[33] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003, 2016. 1, 2, 3, 5, 6, 7

[34] David Nilsson and Cristian Sminchisescu. Semantic video segmentation by gated recurrent flow propagation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6819–6828, 2018. 3

[35] Sebastian Ruder12, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. Latent multi-task architecture learning. In *AAAI*, 2019. 3, 5

[36] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European Conference on Computer Vision*, pages 746–760. Springer, 2012. 2, 6

[37] Trevor Standley, Amir Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? In *International Conference on Machine Learning*, pages 9120–9132. PMLR, 2020. 3

[38] Gjorgji Strezoski, Nanne van Noord, and Marcel Worring. Many task learning with task routing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1375–1384, 2019. 2

[39] Raphael Tang, Weijie Wang, Zhucheng Tu, and Jimmy Lin. An experimental analysis of the power consumption of convolutional neural networks for keyword spotting. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5479–5483. IEEE, 2018. 6

[40] Marvin Teichmann, Michael Weber, Marius Zoellner, Roberto Cipolla, and Raquel Urtasun. Multinet: Real-time joint semantic reasoning for autonomous driving. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1013–1020. IEEE, 2018. 2

[41] 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. 3

[42] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11534–11542, 2020. 3

[43] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018. 3

[44] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 3

[45] Fanyi Xiao and Yong Jae Lee. Video object detection with an aligned spatial-temporal memory. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 485–501, 2018. 3

[46] Dan Xu, Wanli Ouyang, Xiaogang Wang, and Nicu Sebe. Pad-net: Multi-tasks guided prediction-and-distillation network for simultaneous depth estimation and scene parsing. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 675–684, 2018. 2

[47] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. *arXiv preprint arXiv:1805.08318*, 2018. 3

[48] Zhenyu Zhang, Zhen Cui, Chunyan Xu, Yan Yan, Nicu Sebe, and Jian Yang. Pattern-affinitive propagation across depth, surface normal and semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4106–4115, 2019. 2

[49] Ling Zhou, Zhen Cui, Chunyan Xu, Zhenyu Zhang, Chaoqun Wang, Tong Zhang, and Jian Yang. Pattern-structure diffusion for multi-task learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4514–4523, 2020. 2

[50] Xizhou Zhu, Yuwen Xiong, Jifeng Dai, Lu Yuan, and Yichen Wei. Deep feature flow for video recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2349–2358, 2017. 1, 3

[51] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V Le. Learning transferable architectures for scalable image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8697–8710, 2018. 6